Patterns of Patterns: A Methodological Reflection on the Future of Design Pattern Methods

JOSEPH CORNELI, ALEX MURPHY, Oxford Brookes University
RAYMOND S. PUZIO, LEO VIVIER, Hyperreal Enterprises, Ltd.
NOORAH ALHASAN, LBJ School of Public Affairs, The University of Texas at Austin
VITOR BRUNO, Milestone English
CHARLOTTE PIERCE, Pierce Press
CHARLES J. DANOFF, Mr Danoff's Teaching Laboratory

This paper shows how we combine and adapt methods from elite training, future studies, and collaborative design, and apply them to address significant problems in social networks. We focus on three such methods: we use Project Action Reviews to implement social perception, Causal Layered Analysis to implement social cognition, and Design Pattern Languages to implement social action. We present the results of two studies: firstly, we use Causal Layered Analysis to explore the ways in which the design pattern discourse has been evolving. Secondly, to illustrate the three methods in combination, we develop a case study, showing how we applied the methods to bootstrap a distributed cross-disciplinary research seminar. Building on these analyses, we elaborate several scenarios for the future use of design patterns in large-scale distributed collaboration. Our case study suggests ways in which progress could be made towards realizing these scenarios. We conclude that the combination of methods is robust to uncertainty, insofar as they support adaptations as circumstances change, and incorporate diverse perspectives. In particular, we show how methods drawn from other domains enrich and are enriched by design patterns; we believe the analysis will be of interest to all of the communities whose methods we draw upon.

Categories and Subject Descriptors: D.3.3 [**Programming languages**]: Language Constructs and Features—*Patterns*; H.5.3 [**Information Interfaces and Presentation**]: Group and Organization Interfaces—*Collaborative computing*; K.4.0 [**Computing Milieux**] Computers and society

Additional Key Words and Phrases: Design Patterns, Pattern Languages, Action Reviews, Futures Studies, Causal Layered Analysis, Emacs, Free Software, Peeragogy, Climate Change, Innovation, Anticipation

ACM Reference Format:

Joseph Corneli, Alex Murphy, Raymond S. Puzio, Leo Vivier, Noorah Alhasan, Vitor Bruno, Charlotte Pierce, and Charles J. Danoff. 2021. Patterns of Patterns: A Methodological Reflection on the Future of Design Pattern Methods. HILLSIDE Proc. of Conf. on Pattern Lang. of Prog. 28 (October 2021), 23 pages.

Our paper presents a novel synthesis of existing methods from the fields of Learning Management, Future Studies, and Design Patterns. We show how these methods complement each other, and combine holistically into a coherent framework for collaborative design.

The contribution of the paper is simultaneously theoretical and practical. As a stepping stone to the fully integrated method introduced here, we draw on an approach from Future Studies to analyze existing Design Pattern literature and practices, and develop several scenarios that characterize potential directions for the future development of design pattern methods. Additionally, we present a practical case study in the new combined methodology. A limitation of the work presented here is that, so far, we have only anecdotal evidence for the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission. A preliminary version of this paper was presented in a writers' workshop at the 28th Conference on Pattern Languages of Programs (PLoP). PLoP'21, OCTOBER 5–7, Virtual Online. Copyright 2021 is held by the author(s). HILLSIDE 978-1-941652-17-6

method's efficacy. However, we explain why we think further experiments with this approach will be useful both to design pattern thinkers, and other communities. Indeed, we contend that an approach along the lines developed here is needed for tackling the world's largest problems.

1. BACKGROUND

In 1999, the architect Christopher Alexander discussed his work with an audience of programmers, sharing his vision of a synthesis of architecture and computer science that could build towards the generation of a "living world" [Alexander 1999]. Considering the existential risks we face, we may take these remarks as more than just a metaphor.

Design patterns offer a point of entry into this vision. We begin by recalling that Alexander thought about patterns in both a fundamental and a methodological sense.

> *As an element in the world*, each pattern is a relationship between a certain context, a certain system of forces which occurs repeatedly in that context, and a certain spatial configuration which allows these forces to resolve themselves.
>
> *As an element of language*, a pattern is an instruction, which shows how this spatial configuration can be used, over and over again, to resolve the given system of forces, wherever the context makes it relevant. [Alexander 1979, p. 247] (our emphasis)

Patterns in the first sense are basically physical in nature. *Design* comes into play with the second sense. Leitner summarized how this is meant to work: "Patterns are shared as complete methodic descriptions intended for practical use by experts and non-experts" [Leitner 2015]. Already there are a number of practical texts that use patterns (in the second sense) to talk about patterns (again in the second sense): they share methods that aid in discovery, writing, workshopping, and the broader application of design patterns. By contrast we develop a more fundamental analysis, and use this to work towards a new level of practicality.

Alexander's hopeful stance currently comes up against complex global crises. How are we to understand design patterns in this context? Stephen Batchelor writes: "If I am to take this crisis with the seriousness I feel it deserves, then I need to align my thoughts and actions. I require a coherent worldview to provide a rational and ethical foundation for my behavior" [Batchelor 2020]. Pattern methods could fill part of this need. It would be overly simplistic to see in patterns only evidence of a "technical mindset" (*ibid.*), embodied only in a growing repository of technical fixes. Understood in their fundamental sense, patterns can help to articulate "forms of collective action that can respond to the climate emergency that threatens life on Earth" (*ibid.*). Now the relevant forces are no longer simply physical, but are socially distributed and culturally determined.

However, outstanding criticisms show that design pattern methods have not yet reached their envisioned potential [Dawes and Ostwald 2017]. As a step towards that realisation, we now consider two additional methods, and revisit the core features of the design pattern method. The specific selection of methods is grounded in what we have found practical in our shared projects, though for context we provide brief pointers to other methods that serve similar purposes.

1.1 Project Action Review

The US Army developed a methodology called the *After Action Review* [AAR], which they use in training elite soldiers [US Army 2002]. Conducting an AAR facilitates learning from past experience, to generate better future performance. The method has also been used effectively in business settings [Darling et al. 2005]. AARs can be used to assign responsibility when things go wrong, and can help people figure out how to do better next time. As such, the AAR shares common ground with the DAILY SCRUM and SPRINT RETROSPECTIVE [Sutherland 2019] patterns from the Scrum methodology. However, it does not have the product orientation of Scrum.

In a distributed peer-to-peer collaboration, we wanted an adaptation of the AAR which would make it more open ended and horizontal in nature. We came up with the following template:

(1) Review the intention: what do we expect to learn or make together?
(2) Establish what is happening: what and how are we learning?
(3) What are some different perspectives on what's happening?
(4) What did we learn or change?
(5) What else should we change going forward?

When we fill in the template, we call it "doing a *PAR."* As an acronym, "PAR" has stood for various things over the years—Paragogical Action Review, Peeragogical Action Review, Project Action Review—but we like PAR as a stand-alone term. Allusively, it brings to mind the corresponding concept of *par* in golf, and helps give us a sense of how we are doing at any given point in time.[7] Like the Army, we typically use PARs retrospectively ("what *did* we expect to learn or make together?"). However, PARs can also be applied to look forward (to prePARe, so to say) as a way to scaffold anticipation by "remembering the future" [Arnkil 2008]. In that case, item (5) can be expanded to include a number of alternative scenarios.

1.2 Causal Layered Analysis

Sohail Inayatullah developed Causal Layered Analysis (CLA) [Inayatullah 1998, 2004] as a research methodology for examining a topic of concern at four layers that he refers to as the *litany*, *system*, *worldview* and *myth*. Part of the reason to carry out such an analysis is that there are different kinds of causes, ranging from immediate events to deep-seated cultural beliefs. Inyatullah's work draws on his scholarship of P. R. Sarkar:

> For Sarkar, there have been four historical ways humans have dealt with their physical and social environment: either by being dominated by it, by dominating it through the body, dominating it through the mind, or dominating it through the environment itself. [Inayatullah 1999]

In developing a CLA, none of the four layers is privileged over the others, nor are they examined in isolation. Rather, one moves between them, examining how they relate to one another. One can then integrate these insights to form a more comprehensive basis for understanding what is happening in the present and for anticipating the future. Table I describes each of the four layers according to the following schema:

—**Contents**: *What is found in this layer?*
—**Analysis**: *Techniques for analysis of this layer.*
—**Literature**: *Instances of texts which are typically operative at this layer.*

Clemens [2020] relates CLA to the work of Gregory Bateson: specifically, he finds an analogue in Bateson's notion of a "pattern of patterns" [Bateson 1979] which interrelates phenomena across disparate domains.

1.3 Design Pattern Languages

The two senses of 'pattern' mentioned above—'As an element in the *world*. . . ' and 'As an element of *language*. . . '—are mirrored within the concept of a design pattern. Like an ellipse, the design pattern has two main foci: context and community.

—*Context* shapes and constrains the type of activity which is being considered, be it designing a building, writing software, or something else.
—*Community* encompasses the stakeholders—experts and non-experts alike—who are involved with or otherwise affected by a particular project.

Integral to the basic concept of a design pattern is a third feature that describes the interaction of the community and the context. The community uses the pattern to overcome some real or potential *conflict* that they experience within this context. It bears emphasis that the community is not assumed to be homogeneous, and, indeed, this

Litany	
Contents:.	Observable facts, events, and quantitative trends.
Analysis:.	Minimal processing of data.
Literature:.	News reports, tax filings, chit-chat.
System	
Contents:.	The social, economic, political, and historical forces which shape events.
Analysis:.	Technical explanations and interpretation of data within a given paradigm.
Literature:.	Editorials and policy institute reports.
Worldview	
Contents:.	Core values and attitudes which motivate choices and actions.
Analysis:.	Uncover deep assumptions and study the mental and linguistic constructs which undergird how people interact with each other and their surroundings. Compare and critique paradigms and discourses.
Literature:.	Works of philosophy and critical theory.
Myth	
Contents:.	The symbols and tales which give meaning to life.
Analysis:.	Study symbols and narratives, and the myths and rituals within which they participate.
Literature:.	Poetry, art, anthropology, Jungian analysis.

Table I. : Overview of the layers in Causal Layered Analysis

may be part of how the conflict is experienced; i.e., it need not be the case that all members of the community share the same experience or view of the context, nor that they are all uniformly affected by the circumstances arising therein. The conflict is also referred to as a *problem*; its resolution is described as a *solution*. Alexander and Poyner emphasized that 'design' is not needed when the conflict can be resolved in an obvious or straightforward manner. For example, you typically would not need a design process surrounding *sitting in a chair*,

We might say that the design pattern carries with it a fragment of irreducible complexity. This perspective may or may not be surprising. Early on, Alexander described the need for patterns when things get complex [Alexander 1964]. He specifically focuses on what could be called "horizontal" complexity, a situation where there are a lot of moving parts and relations between them. Methodologically this is elaborated with the notion of a *pattern language*. Pattern languages have a property of unfolding, from more general to more specific. However, they do not necessarily cover deeper forms of "vertical" complexity, where there are deep historical or ontogenetic causes, feedback loops, or complex conceptual issues which are not readily expressible in design-pattern-theoretic terms. Let's have another look at these issues by way of two contrasting metaphors.

The first metaphor comes from Christian Kohls, who proposed to treat each design pattern as a journey: "a path as a solution to reach a goal" [Kohls 2010]. In this metaphor, design patterns are understood to have an initial condition and an end condition, defined within some context. The context also associates a cost to traversals of paths. There are several associated problems: the elementary problem is to traverse the terrain and travel from the start state to the end state. The next problem is to do this at low cost. The third problem is to find a reliably repeatable way to do this. A fourth problem is to describe the process in such a way that the path can be traversed by others.

The second metaphor comes from Joseph Campbell, who described an "archetypal pattern" [Shalloway and Trott 2005], one that can be found embedded in myths and stories across diverse cultures and historical periods. The "hero's journey" is also described with a path [Campbell 1949], however, in this case the path runs in a

circle, and the journey focuses on the transformations of the hero who traverses it. Although an account of the journey can be shared, traversal is effectively single-use. The cost is typically "high." Nevertheless, once a myth or metaphor is established in a shared narrative, the journey can be reenacted through ritual or engaged with in other ways that solve a range of social problems [Handelman 1998]. In short, the difference between these two traversal stories suggests that the process of finding "the path that is capable of leading to a good structure" [Alexander 1999] may contain irreducible complexity—even when sharing the information about the path is relatively simple.

2. METHODOLOGY

Each of the three methods described above has distinct use-cases when considered in isolation. Viewed as alternatives, they would have various tradeoffs between them, concerning their ease of use, their ability to generate solutions, the breadth of their applicability, and so on. In this section we put them together as one holistic pattern of patterns. We will use them to scaffold social perception, cognition, and action (Table **??**).

We adopt this tripartite division from classical psychology [Hilgard 1980]; it continues to be relevant in contemporary neuroscience [Teufel and Fletcher 2020; Friston 2013]. Indeed, it is not only as a division, but precisely in characteristic combination that these three factors become important. This suggests a third understanding of 'pattern' as "a dynamic [...] ongoing awareness created and upheld by a mass of ongoing physiological activity" [Iran-Nejad and Winsler 2000]. To study such factors in integration "requires understanding the conditions and laws of construction in mental life" (*ibid.*). So it is, as well, for collective knowledge production. To recapitulate and reframe the methods in these terms:

Project Action Review (PAR) is the sensory element: systematically gathering and verifying observational data. The Project Action Review is structured around five questions which members of a community discuss and answer together. This practice generates a record of an event, as seen through the eyes of the participants. In the moment, the PAR allows us to draw out views which might have gone unstated otherwise. Over time, projects which use PARs improve their chances of staying grounded in reality as circumstances evolve. We can additionally use PARs to help check how effectively we are using other methods.

Causal Layered Analysis (CLA) is the cognitive element: giving organization and depth to the enterprise. The goal of this methodology is to achieve a deep and inclusive understanding by integrating empiricist, interpretative, critical, and actionable knowledge surrounding a topic of concern. Without an integrated understanding, a group runs the risk of getting lost in a muddle of details. CLA can pull information logged in PARs together into a coherent body of self-knowledge. It can also help to surface concerns that might remain implicit in pattern language: for example, CLA could help us understand why we had prized a technological solution to what, upon consideration, turned out to be a fundamentally social issue (or vice-versa).

Design Pattern Language (DPL) is the motor element: orchestrating and scaffolding action. Having carefully analyzed the situation and identified possible solution pathways, we must plot a course of action that accounts for the complexities of the situation. Individual design patterns present solutions to recurring problems: they can be combined with other patterns and adapted to different situations. A DPL is a common language which a community can use to discuss matters of design; it serves as a repository of shared knowledge. The flexibility of DPLs allows the structure to be customized to our particular circumstances as they evolve.

PLACARD: A Synthesis of PAR, CLA, and DPL

We are now in a position to explain how PAR, CLA, and DPL combine into one holistic pattern, in Leitner's sense of a complete methodic description [Leitner 2015]. We will write this down using the classic DPL format: describing the associated *context*, the *problem* denoting a conflict, together with a *solution*. As it happens, the three acronyms introduced earlier can be combined and remixed to provide a title for this pattern.

$$\text{PAR} + \text{CLA} + \text{DPL} = \text{PLACARD}$$

This accurately suggests that the methods need not be run in a fixed order, but are interwoven together.

—**Context**: In the course of working on a project: *we use the PAR to get a sense of our working context*.

—**Problem**: Although we may encounter many difficulties in this context, our effort to understand them faces a central **challenge**, namely the fact that the problems span different layers and scales of complexity, so it can be hard to understand where the difficulties actually come from: accordingly, *we use the CLA to understand and frame the problems and their interconnections*.

—**Solution**: Once we have grasped the problem, we need to elaborate an actionable solution that remains adaptable to ongoing changes in the context: *we use DPL to elaborate the solution (returning to PAR and CLA as needed)*.

Fig. 1. Mnemonic illustration of the PLACARD pattern

Figure 1 provides a mnemonic. The main thing to notice is that using the three methods together can help make the design pattern method practicable. We can use the PAR to move from a context to a “Context”, established and written down. We can use CLA to move from a situation of concern to a situation in which the core “Problem” or “Problems” can be thought about.

The fact that DPL shows up inside of PLACARD may be somewhat concerning. This concern calls for some further brief remarks before turning to a case study of the PLACARD pattern in use.

A solution process can often be decomposed into interrelated subtasks [Alexander 1964; Polya 1945, 1954]. A standard problem-solving methodology is to understand the (sub-)contexts and (sub-)problems in detail—along with their relationships to other parts of the developing decomposition—and on this basis make predictions about the way an intervention could change the overall system. Accordingly, problem solving often goes back and forth between “two different forms of information processing: bottom-up and top-down” [Teufel and Fletcher 2020]. CLA can help with thinking in both directions. Nominally, CLA is an *analytic* tool that decomposes a problematic situation into *layers*, and *causes* operating at and between these layers: in this sense it functions top-down. However, CLA also plays a bottom-up synthesis role. Whereas we could compare the PAR to a tangent vector or derivative that gives a momentary reading of how things are going at a given point in time, CLA can be used to integrate these observations into a plan.

3. RESULTS

We report on two studies traversing different scales. Firstly, we applied CLA to the design pattern literature and practices, with the purpose of scaffolding an examination of the future of the design pattern theory. Secondly, we made practical use of CLA alongside the PAR and DPL methods (i.e., PLACARD as a whole) within a case study of a distributed research seminar.

3.1 Analysis: Design Pattern Language literature and practices

In this study we examined DPL literature and practices in the form of books, articles and conferences. In order to counterbalance the coverage, we included dissenting and critical voices. Here we did not have PARs at our disposal, so the sensory element was provided by the views expressed by previous authors.

In précis, we were interested in the following topics:

—Queries raised by Alexander and his collaborator Bryant, along with a systematic analysis of criticisms of pattern methods collected by Dawes and Ostwald.

—Issues related to how people share and discuss patterns, as well as the changing way in which these discussions have been framed at PLoP.

—The worldview linked with patterns through the lens of mob software and its critiques.

—Symbols and philosophical traditions that enrich our understanding of the context in which Alexander developed his methods.

Taking a deep dive into DPL via CLA allowed us to gain perspectives on how design patterns work. In particular, we have illustrated the complexity that underpins the model. Alexander expands on his metaphysical considerations in his multi-volume work, *The Nature of Order* (*TNO*). Our Analysis shows how some of these more ephemeral-seeming factors are ramified across various layers of the pattern theory.

In particular, working from the myth layer towards the more surface layers: we show that the notion of *wholeness* that Alexander deploys is complex, and that the corresponding theory of emergent order based on this concept is similarly complex and linked to “a tension between independent and conforming tendencies” [VanDrunen 2010]. This tension has—only partially—resolved into a dichotomous relationship between *users* and *designers* of patterns and pattern-linked artifacts. There are however many remaining points of friction, as users of pattern methods run into difficulties [Dawes and Ostwald 2017], and designers are not entirely clear on how to improve the situation.

3.2 Case Study: Emacs Research Group

The Emacs Research Group (ERG) is a transdisciplinary seminar organized around the theme of research in, on, with, and about the Emacs computer program. Although Emacs is best known as a text editor, its extensibility and self-documenting nature make it a more general platform for dealing with symbolic content, and a vibrant site for research into writing and programming. ERG aims to explore topics such as the following:

—How is Emacs *used* to help conduct research in various disciplines?

—What is the *user experience* with Emacs and what is the user community?

—How could the Emacs system *interoperate* with other computer programs?

—How might the *communities* affiliated with Emacs interact with other communities of software users and developers?

—What new *functionalities* would broaden the applicability of Emacs?

ERG meets approximately weekly, sometimes inviting guest speakers. After each meeting, the participants summarize their experience in a PAR. This serves to surface matters of concern and highlights in each session of the seminar. Every six meetings, the techniques of CLA were used to organize and condense the PARs into a

coherent statement of purpose. Finally, Peeragogy design patterns are used to formulate a plan of action informed by this analysis. Thus, all three methods are combined in line with the PLACARD pattern introduced above.

In this study, we were most interested in understanding and exploring the PLACARD method and its potential efficacy; e.g., could we use the PLACARD-associated methods to develop actionable and deliverable designs for new software? After a year of working with the methods together, we presented them at EmacsConf 2021, advocating that other Emacs enthusiasts adapt them for use in creating their own small research groups.

Summary of Findings

We used Causal Layered Analysis to describe the evolution of Design Pattern Language methods in response to criticism, innovation, technical developments, and long-term cultural change. We additionally surface some of the forces and tensions in the discourse. We used Design Pattern Language methods, together with Project Action Reviews and Causal Layered Analysis, to organize a research seminar. This case study shows how the methods can be fruitfully combined, and suggests how others might take up the combined methods (e.g., by substituting 'Patterns' for 'Emacs' as the focus in a future research seminar).

4. DISCUSSION

Informed by the two studies described above, we would like to reflect on why putting the CLA and DPL methods together can make a big difference in practical terms. To do this, we begin by examining a specific problem domain to which CLA and DPL have been applied separately.

Anthropogenic climate change is a situation of major global concern in the early 21st Century. It comes as no surprise that it has been examined separately by proponents of both CLA and DPL. We use this recent history to frame future work building on the case study and analysis developed above.

In an overview of their work on the Cooling the Commons pattern language, Cameron Tonkinwise and Abby Mellick Lopes write:

> A design pattern is first an observation: "People in that kind of designed situation tend to do this sort of thing". It is then possible to design an intervention that redirects those tendencies. If that intervention succeeds, it can become a recommended pattern to help other designers: "If you encounter this kind of situation, try to make these kinds of interventions" [Tonkinwise and Lopes 2021].

They amplify the 'ethical' aspect of their thinking:

> . . . the patterns we are talking about, context-specific interactions between people and things, are more like habits. They are tendencies that lead to repeated actions.

The 41 patterns they have developed include examples like THE NIGHT-TIME COMMONS, which:

> . . . might shift daytime activities into cooler night times. Some places already have these patterns: night markets and night-time use of outdoor spaces. If locally adapted versions of these patterns encourage people to adopt new habits, other patterns will be needed. These will include, for example, ways to remind those cooling off outdoors in the evening that others might be trying to sleep with their naturally ventilating windows open. Such interlinked patterns point to the way pattern thinking moves from the big scale to the small.

Reading this, we were concerned that, while the Cooling the Commons patterns do acknowledge *horizontal* complexity—namely, through interlinked patterns—the process does not deal with the *vertical* complexity coming from the fact that diurnal rhythms are deeply embedded in biology and culture. People have cultural beliefs about the activities that are appropriate for different times of day. Public and domestic rituals are organized about the daily cycle. Times of day have symbolic associations. As far as we could tell, these authors focused on more or less technical issues at the systems level, and did not acknowledge these issues at the worldview and myth

levels. A more comprehensive approach might, for instance, re-examine rituals to see which of them relate to the phenomenon of sunrise versus the act of getting up and starting the day, and then figuring out how to adapt these rituals to a new schedule. A suitable research strategy might be to study how practices changed in the past, as with the introduction of industrialization and its clockwork regimentation of the day.

Meanwhile, Heinonen and coauthors [Heinonen et al. 2017] describe a CLA game that explored four different scenarios in small groups. The four scenarios were “Radical Startups”, “Value-Driven Techemoths”, “Green DIY Engineers” and “New Consciousness”. As groups worked through the CLA for each scenario, they developed a range of new ideas. We wondered, how might these CLA-linked reflections collate against the Cooling the Commons patterns? Might players of the CLA game have spotted ways in which the patterns would conflict with deeper values—or ways in which they might be exploited to cause chaos [Friction 2016]?

Broadening our exploration of how design patterns relate to futures studies, we note that Schwartz [1996] (Appendix, pp. 241-248, *viz.*, his “Steps to Developing Scenarios”) described a process that follows an outline that is strikingly similar to a design pattern template. Both Alexander and Schwartz advocate the identification of driving forces in a context. However, unlike Alexander, Schwartz does not intend to resolve conflicts between the forces within a harmonizing design. On the contrary, the aim in the scenario development method is to understand how these forces might evolve and lead to the further diversification of scenarios. To simplify the process, some scenario planners reduce the number of forces to the two most important. For example, the scenarios of Heinonen et al. are organised along two axes: one spanning different degrees of integration between peers, the other, different levels of ecological awareness.

With these reflections in mind, we came up with four scenarios for the future development of design patterns. The four scenarios offered here are inspired by the four CLA layers: litany, system, worldview, and myth.

4.1 PLACARD becomes transferable and computational

With further work, PLACARD could become a more refined but nevertheless easy-to-use instrument for gathering, organising, and sharing data on problem solving in social networks. Building on this collected data, we could take further steps to develop computational models of future imaginaries. Patterns have previously been discussed in explicitly computational terms—however, that direction of work so far remains mostly at the level of a proposal [Alexander 1999; Moran 1971], with only limited discipline-specific uptake (e.g., [Jacobus 2009], [Oxman 1994], [Taibi and Ngo 2003]). Could this change, to generalize the kinds of patterns that can be computed with, and make them interoperable? Polya had already been writing about *patterns of plausible inference* the year that Alexander started his undergraduate degree in Cambridge [Polya 1954]; four years later Polya’s student Allen Newell was beginning to think about how to model the inference process computationally [Simon and Newell 1958; Newell 1983]. In the domain of economics, Ostrom-style institutions are analogous to design patterns ([Ostrom 2009], p. 11). Recent work looks at how descriptions of such institutions can be extracted from text [Rice et al. 2021]. Could this line of thinking be extended, so that other similar kinds of patterns could be recognized where they appear? Could the extracted descriptions be used directly in computations? One fruitful strategy might be to think of design patterns as *conceptual blends* [Corneli et al. 2018], which can be given a computational interpretation [Schorlemmer and Plaza 2021]. For example, the COMMUNITY LIBRARY pattern from the Cooling the Commons pattern language blends a learning space with a cool refuge. Could such complexities be reasoned about computationally?

4.2 Pattern language authoring communities move to free/libre/open source licensing

In the field of policy, *resilience* describes a society’s ability to recover after a shock; whereas *adaptive capacity* describes its ability to move to a new state [Thonicke et al. 2020; Magnan 2010]. This ability is, in turn, linked with the health and adaptivity of the society’s institutions [Fidelman et al. 2017]. Free/Libre/Open Source licensing is one possible institutional innovation in the way design patterns are used. Widespread adoption of new licensing arrangements could support social learning, which could, in turn, boost adaptive capacity [Thi Hong Phuong

et al. 2017]. As an example of work heading in this direction, Mehaffy and coauthors collaborated with Ward Cunningham to make their book *A New Pattern Language for Growing Regions* [Mehaffy et al. 2020] into a wiki, npl.wiki, which is licensed under CC BY-SA 4.0. Will other pattern developers follow suit and move to open licensing—and suitable infrastructures for working with open contents? We can also ask: what about other kinds and qualities of openness? A 'copyleft' license is not a panacea for all ills [Krowne and Puzio 2006], and would not on its own make the pattern theory and methods fully open in all the ways that matter. Nevertheless, grappling with the challenges around licensing and related considerations could serve as a rallying point for the pattern community.

4.3 Patterns empower individuals and communities

As we've seen in our work with Emacs and Peeragogy (and previously with the online community PlanetMath [Krowne 2003; Corneli 2014]) projects need more than just access to source code in order to thrive. We see a link to the topic of reproducible research. Above and beyond the immediate technical considerations [Sandve et al. 2013], the process of doing science is "reproducible" if the methods are teachable to others. The Literate Programming paradigm can help with this. At the same time, collaboration across different skill sets is challenging; large scale problems like adapting to climate change seem to require such collaboration, and almost certainly won't be solved if we carry on doing business as usual. In the Minnesota 2050 project, participants were selected from a variety of professions and leadership roles to produce scenarios for energy and land use, and combined modeling with scenario planning [Olabisi et al. 2010]. Actually solving large-scale problems in interdisciplinary teams will require new thinking and additional tools: to bridge between the viewpoints of, e.g., professional futurists, programmers, data scientists, local farmers—and to draw on the insights of citizen scientists [Wildschut 2017]. CLA and PLACARD can help; additional patterns of patterns will be needed to work fluently across domains, levels, and spheres of endeavour.

4.4 Patterns facilitate economic empowerment

Access and meaningful participation are serious matters of concern in our current technological culture [Unger 2019]. Patterns could become the basis of widespread capabilities and substantial opportunities [Robeyns and Byskov 2021], shared not only by an elite group of hackers or a few highly-paid rockstars, but by everyone. Patterns have been used to describe soft skills that are useful for aspiring programming professionals [Hoover and Oshineye 2009]: however this falls far short of reforming the tech sector or our broader relationship to technology. As a related example from the world of literature, Herman Hesse's novel *The Glass Bead Game* imagines a society in which the elite community of scholars studying abstract patterns form a strong hierarchy, and remain mostly out of touch with the practical realities experienced by outsiders. The story serves as a warning: we must proceed with caution when we seek to bridge practice and theory. When reflecting on futures-oriented discourses, Slaughter described a spectrum: "participatory and open at one pole and closed (or professionalised) at the other" [Slaughter 1989]. Are pattern authors prepared to work with widespread stakeholders to study the forces that shape the economy and society?

5. CONCLUSION

In 1999, the architect Christopher Alexander discussed the future of design patterns with an audience of programmers [Alexander 1999]. We revisited this topic from a methodological perspective. Building on the analysis and case study we developed, we discussed several scenarios for the future of design pattern methods. Our vision for change is that these four scenarios will be given serious thought by other members of the patterns community. Our Case Study suggests concrete ways in which progress might be made towards realizing these scenarios, or, indeed, if warranted, towards rejecting them and developing another vision.

Progress will become measurable through markers of debate and dialogue between the different communities whose work we have drawn upon, and potentially, through trial-and-error uptake or adaptation of the methods

we've described. We are certainly not the only people to think about systems and futures: what is distinctive about this paper is that we've connected these domains with design pattern terminology and methods. Some potential implications of that connection are embedded in the scenarios outlined above.

In the present exploration, we began by thinking about patterns from a fundamental perspective: *patterns as elements in the world*. Some patterns repeat in space, some in time, some in both space and time; think of a tiling, a beat, a wave. However, patterns cannot repeat exactly or forever: their elements are subject to spatial or temporal displacement, and other forms of variation. We need suitable abilities—and methods—to perceive and work with patterns. The methods we used in this paper were the Project Action Review (PAR), Causal Layered Analysis (CLA), and Design Pattern Languages (DPL)—though other methods that fulfill the same basic purpose could be used without significantly changing the overall import of what we say here.

—By using the PAR (or another sensory method), we are able to identify recurring themes.

—Then, by using the CLA (or another cognitive method), we are able to organize these repeating themes in a structure that exposes the underlying trends, causes, and potential terminating states.

—With DPL (or another motor method) we can make what we have learned actionable.

A contribution of this paper was to link the sensory/cognitive/motor theory with design patterns, which we illustrated by combining PAR, CLA, and DPL into the PLACARD pattern.

A key experiential finding that emerged from our joint work on this paper is that we need to take health, well-being, and time for reflection into account, even more than we worry about accomplishments or deliverables, or we are likely to run out of energy before we get very far. Outcome-based planning needs to be balanced against more intention-based planning that takes human factors into account. In future work, we hope to explore the rich factors that contribute to health and adaptivity in groups and networks. The methodological contribution of this paper can help to scaffold that process. Alongside the PLACARD method, we have started to develop a set of design requirements for software that can help people apply the method in their working contexts.

The limitations of the research presented will be clear. Our case study examined a small and well-integrated population, which is moreover, a subset of the authors of the present paper. While the case study provides a proof of concept for the applicability of the methods, in other settings, some or all of the methods might not be needed, or accepted without significant alteration. Despite the limitations of this study when it comes to external validity, evidence to support the use of the integrated PLACARD pattern may accumulate rapidly, if the method proves useful for other transdisciplinary collaborations. By using and negotiating around the same DPL, disparate groups may get a useful external source of validation. We hypothesise that PLACARD could quickly make the work of Learning Management, Future Studies, and Design Pattern practitioners more robust.

We can imagine the methods we've discussed being readily applied, initially, in workshops or other small groups. For example, if we were to gather a group of citizens, experts, and other stakeholders to talk together about the city, they could use the methods we've described to surface issues, rehash concerns, and sensemake together. "Anticipatory Social Science" is a broader term for this kind of work. Moreover, working with methods that distribute perception, cognition, and action, we may become more comfortable with uncertainty, and better able to support innovation. Especially when grappling with massive problems like climate change, we will need to develop a transformative set of tools and methods [Miller 2018]; and beyond this, a coherent worldview, and perhaps even new myths. Citizen science has a potentially important role to play here [Wildschut 2017], alongside other forms of participatory culture. The methods we described may help to support widespread engagement, and coordinate activities across scales: from individuals, to communities and social movements. As Nietzsche said, "the more affects we are able to put into words about a thing, the more eyes, different eyes, we can use to observe one thing, the more complete will our 'concept' of this thing, our 'objectivity,' be" [Nietzsche 2017] (p. 128).

Acknowledgements

We thank our PLoP 2021 shepherd Michael Mehaffy for supportive comments and productive discussion. We thank the participants in our PLoP Writer's Workshop group, and Mary Tedeschi and her students in Information Systems Development at Baruch CUNY for additional comments on the Conference draft. This manuscript is distantly derived from submissions to the Connected Learning Summit 2018, Anticipation 2019, and the *Futures* journal. We thank Paola Ricaurte, Analua Dutka-Chirichetti, Hermano Cintra, Lisa Snow MacDonald, and Verena Roberts—coauthors of these earlier submissions—for helping to shape our thinking here. We thank Claire van Rhyn for bringing the Anticipation conference to our attention and thereby putting us in touch with the field of futures studies. We appreciatively acknowledge feedback on the ideas in the paper that we received as they were developing, from Roland Legrand, Charles Blass, Stephan Kreutzer, Giuliana Marques, Cris Gherhes, Qiantan Hong, Cameron Ray Smith, David O'Toole, Rebecca Raper, Steve Corneli, and Elizabeth Cheetham.

The image appearing in Figure 1 was made available under the terms of CC Zero by User:Lapsed Pacifist at `https://commons.wikimedia.org/wiki/File:Shell_to_Sea_placard.jpg`.

Supplementary Material

A. SUPPLEMENT: ANALYSIS: CLA APPLIED TO DESIGN PATTERN LANGUAGE LITERATURE AND PRACTICES

In developing this analysis we are aided by several additional methods from the Poststructural Futures Toolkit [Inayatullah 1998]. We will refer to the individual methods from this toolkit with their names rendered in all-caps.

A.1 Litany: Understanding data, headlines, empirical world (short term change)

The first layer in CLA is the **litany layer**: it describes the problems that people are well familiar with. In the case of the design patterns discourse, this level includes—in particular—the familiar kinds of conflict-based problems that are described in patterns and discussed at PLoP, along with higher-order problems of application, and debates about these (e.g., ranging from Christopher Alexander's "ENTRYWAY TRANSITION" pattern to his remarks about how people who attempted to apply his methods ended up placing "alcoves everywhere", etc.). This layer is sometimes also referred to as the **problem level**: in the patterns discourse, problems abound. Indeed, one of the core attributes of the pattern community is that it is not only comfortable with problems but that it actively seeks them out with a 'problematizing' discourse.

Not all of the well-known and discussed problems have been solved. For example, 'Alexander's Problem', as described by his collaborator Greg Bryant, is that:

> . . . despite all of the tools he created, his penetrating research, his many well-wrought projects, and his excellent writing, he did not manage to grant, to his readers, the core sensibility that drove the work. He also did not organize the continuance of the research program that revolves around this sensibility. [Bryant 2015]

Attempts to work out a practical solution to this problem are developing. Coming at the same basic issue from a more visionary standpoint, Alexander framed this query for the programmers who were using pattern methods at the turn of the millennium:

> What is the Chartres of programming? What task is at a high enough level to inspire people writing programs, to reach for the stars? [Alexander 1999]

More recently, Dawes and Ostwald [2017] develop an elegant taxonomy of criticisms of the pattern method. In outline, their taxonomy covers criticisms at the following three layers:

> *Conceptualization.* Ontology, Epistemology
> *(e.g., "Rejecting pluralistic values confuses subjective and objective phenomena")*

Development and documentation. Reasoning, Testing, Scholarship
(e.g., "The definitions of 'patterns' and 'forces' are inexplicit")

Implementation and outcomes. Controlling, Flawed, Unsuccessful
(e.g., "Patterns disallow radical solutions")

By showing how the criticisms relate to one another, Dawes and Ostwald begin to develop a GENEALOGY at the level of critical perspectives. The critiques they examine show that there is not just one pattern discourse, but many. In a parallel work the same authors analyze the structure of Alexander's classic text, *A Pattern Language* (*APL*) and develop three alternative perspectives on *APL*'s contents, which they refer to as the **generalized**, **creator**, and **user** perspectives [Dawes and Ostwald 2018]. These perspectives amount to different techniques for REORDERING KNOWLEDGE. We will elaborate at the next level.

A.2 System: Systemic approaches and solutions (social system)

The **system** layer is typically understood in terms of the **social phenomena** that cause the problems at the litany layer to emerge (along with their familiar solutions). In the original setting in which patterns developed, this layer would have included causes such as more people living in cities, combined with the possibility of developing a more community-driven approach to design using contemporary technologies. In short, at this level, we examine where the familiar problems come from.

Dawes and Ostwald's [2018] central finding is that many patterns in which Alexander had medium or low confidence in fact occupy a relatively central position in *APL*'s graph:

> . . . the patterns which are most likely to be encountered by designers – are most easily accessed, or provide greatest access to other patterns – might be those which Alexander acknowledged were incapable of providing fundamental solutions to the problems they addressed.

This means that novice users could be expected to encounter problems in application of *APL*'s patterns: "despite its often authoritative and dogmatic tone, Alexander's text was framed as a work in progress, rather than a definitive design guide" (p. 22). Dawes and Ostwald suggest that their analysis could point to "prime opportunities to continue the development of *A Pattern Language*" (p. 21).

Broadly put: there have been some attempts at creating systematic archives of patterns [Köppe et al. 2016; Inventado and Scupelli 2017], but these efforts haven't always garnered significant buy-in. Importantly, the first-ever Wiki was developed in connection with a platform for developing, sharing, and revising pattern languages [Cunningham and Mehaffy 2013].[9] However, there was a distinction between the discussions and the finished patterns. In the 2013 retrospective, Ward Cunningham and coauthor Michael Mehaffy write:

> The original wiki technology functioned in a direct open-source mode, which allowed individuals to contribute small pieces to incrementally improve the whole. (*ibid.*)

This is true if by "open source" we understand what you see when you click Edit—but the statement could be misleading relative to contemporary usage, which is often linked with the Open Source Initiative's definition, which centers on the premise that "Open source doesn't just mean access to the source code." On the `c2` wiki, licensing was restrictive. Discussions were to take place in "letters and replies" rather than revision or annotation of the published patterns; rights associated with the finished patterns were closely guarded.[9]

Although Wiki technology could in principle have been a site for ongoing DECONSTRUCTION of patterns, this hadn't happened on `c2`. This is itself interesting and worth deconstructing a bit. Notably, there were only *four* published "letters and replies". Unfortunately, we could not find a public archive of the design patterns mailing list where further discussions took place. This suggests certain factors of contingency in the development of the discourse. Over the years, some of these concerns have been addressed—e.g., through the introduction of Federated Wikis and new licensing models—and other issues and concerns came to the fore.

Dawes and Ostwald's [2018] remarks on multiple perspectives on pattern languages resonate Jenifer Tidwell's charges against the Gang of Four:

> . . . the reality of a software artifact that the developer sees is not the only one that's important. What about the user's reality? Why has that been ignored in all the software patterns work that's been done? Isn't the user's experience the ultimate reason for designing a building or a piece of software? If that's not taken into account, how can we say our building – or our software – is "good"? — "The Gang of Four Are Guilty" (`http://www.mit.edu/~jtidwell/gof_are_guilty.html`)

Notice that the *user* of the designed artifact has entered the story as a different figure from the user of the pattern language, whom we met above. Tidwell's critique suggests at least a couple ALTERNATIVE PASTS AND FUTURES: e.g., what if the end-user had been placed at the center the whole time? Alternatively, what if the primary focus of patterns was to facilitate interaction between different stakeholders? The fact that Tidwell's book [Tidwell 2010] and an essay by Jans Borchers [2008] which drew inspiration from her critique both have over 1000 citations on Google Scholar shows that Tidwell's perspective has been impactful. To get a sense of how the pattern community may have been informed by this critique—alongside other related trends and concerns—we can look at how the Writers Workshops at PLoP have evolved over time. In Table II, a selection of titles of workshop sessions show how the focus of PLoP evolved from primarily 'programming' oriented to a much broader contextual view over time. Indeed, by 2019, the focus is almost exclusively 'contextual'. The way the themes under discussion have evolved brings to mind the layers of CLA.

1997	**2011**	**2015**	**2019**
Architecture	Architecture	Pattern Writing	Group Architecture
Roles and Analysis	Design	Software Architecture & Process	Culture
People and Process	Information	Cloud & Security	Meta
Domain Specific Techniques	People	Innovation & Analysis	Education
OO Techniques	Pedagogy	People & Education	
Non-OO Techniques			

Table II. : Evolution of PLoP Writers Workshop topics in selected years: CLA in the wild?

A.3 Worldview: ways of knowing and alternative discourse

The next layer comprises **worldviews** (e.g., Alexander's view that "There is a central quality which is the root criterion of life and spirit in a man, a town, a building, or a wilderness").

The situation with licensing on `c2` is particularly interesting in light of Alexander's perspective that *APL* was a "living language". In principle, Wiki technology might have presented the opportunity to realize this vision fully for the first time, in a virtual setting. Wiki technology did become widely influential when it was combined with a free content license on Wikipedia (originally GNU FDL, later CC-By-SA).

Fast-forwarding to the present day, Christopher Alexander's website `patternlanguage.com` writes about The Struggle for People to be Free—but it is not referencing freedom in the GNU sense.

In 1979 he was concerned: "Instead of being widely shared, the pattern languages which determine how a town gets made becomes specialized and private." In 2021, *APL* itself is only legally available for subscribers or for people who purchase a paper copy of the book. (Or through a library!) Of course, like many famous texts it can also be obtained extra-legally for download as a PDF: but that format does not afford downstream users the opportunity to collaborate on the text's further development.

Gabriel and Goldman talk about sharing and gift culture in their essay Mob Software: The Erotic Life of Code. This reference suggests another reason why sharing knowledge in non-editable formats can be problematic. Their

primary source on gift culture is Hyde [2019], who talks about what happens when exchange items are taken out of the gift exchange culture and put in a museum:

> A commodity is truly "used up" when it is sold because nothing about the exchange assures its return. The visiting sea captain may pay handsomely for a Kula necklace, but because the sale removes it from the circle, it wastes it, no matter the price. Gifts that remain gifts can support an affluence of satisfaction, even without numerical abundance. (*ibid.*, Chapter 1, p. 29)

Gabriel and Goldman reference the open source community—but not the free software community, so we will follow their usage here—as the origin of Mob Software.

> Because the open source proposition asked the crucial first question, I include it in what I am calling "mob software," but mob software goes way beyond what open source is up to today. [Gabriel and Goldman 2000]

That question is: "What if what once was scarce is now abundant?" It is well known that the PLoP conference series builds on this idea: it includes shepherding and workshops [Gabriel 2002] as well as games, informal gifts, and other measures that aim to create a sense of psychological safety: all features that make PLoP a space where 'failure' is OK and even celebrated, as per Mob Software. The essay develops its own criticisms of open source, e.g., "the open-source community is extremely conservative" and forking happens rarely. (Five years later, with the creation of Git, a certain form of forking became more typical.) Resonating with Tidwell's critique from above:

> One difference between open source and mob software is that open source topoi are technological while mob software topoi are people centered.

On a technical basis, Gabriel's vision sounds a lot like today's world of *microservices*. While his vision hasn't fully come to pass—for example there are still many services with proprietary source code—nowadays many big companies are also big proponents of open source. Here we can notice that Gabriel was employing a technique of imagining ALTERNATIVE PASTS AND FUTURES, e.g., he imagined a future in which:

> Mentoring circles and other forms of workshop are the mainstay of software development education. There are hundreds of millions of programmers.

We would like to dig somewhat deeper into the foundations of the worldview that Gabriel puts forth in this essay. Usefully, an article by VanDrunen "traces the source of Gabriel's ideas by examining the authorities he cites and how he uses them and evaluates their validity on their own terms" [VanDrunen 2010]. VanDrunen's critique functions as a (detailed) DECONSTRUCTION of the thinking behind Gabriel's essay. Some key excerpts appear in Table III. It is worth noting that this is by no means a complete critique. As an an example of one direction that we will not have time and space to develop here, some applications of the concept of 'gift culture' have been criticized as hegemonic in nature [Mallard 2019]: should we expect pattern-theoretic, mob, or free/libre/open source software culture to be immune from such concerns? VanDrunen's critique is useful for our purposes not because they provide the last word, but because this criticism points to the importance of considering the deeper layers in developing a concept or approach. There may also be conflicts at these deeper layers.

It is also worth noting that mob software is but one of many diverse visions of the future of programming [Nobel and Biddle 2002]. An embrace of diverse perspectives seems to be a fundamental part of the associated worldview. After all, the primary theoretical model of a computer is termed "universal". Perhaps there is a bit of a paradox or double bind here, insofar as we embrace diverse perspectives just as long as they are compatible with our core tenets. For at least some pattern authors, these include "their love of programs and programming" (*ibid.*). (On this last point, both VanDrunen and Gabriel seem to agree.)

Table III. : Key observations from VanDrunen's critique of Gabriel's "Mob Software" essay

"Kauffman's work is about a rediscovery of the sacred, and it amounts to a proposal of the laws of self-organization as a new deity"
"One thing we find in common with Lewis Thomas's ants, Kauffman's autocatalytic sets of proteins, and the agents inhabiting Sugarscape is that they all lack intelligence."
"In other words, the rules given by Gabriel describe only the conforming aspect of group behavior. In reality, there is a tension between independent and conforming tendencies, and the flock patterns emerge from the interaction between the two."
"His examples of 'mob activity'... the making of the Oxford English Dictionary, cathedral-building, and open source software discussed later—all had oversight, master-planning of some sort."
"There are several distinct senses of 'gift' that lie behind these ideas, but common to each of them is the notation that a gift is a thing we do not get by our own efforts." [Quoting Hyde [2019].]
"Certainly proprietary code is shared property among those working in a corporate development team, but it is not common to the larger community of software developers and users."
"A computer program is not like a poem or a dance in this way; if the programmer is not able to produce something parsable in the programming language or cannot fit the instructions together in a logical way, the program simply will not work."
"Gabriel's own experience may color his perception. He founded a software company that produced programs for Lisp development and which went bankrupt after 10 years."
"Moreover, if Gabriel means to suggest that these programming languages or models could have made programming more accessible to the masses lacking technical skill, it is quite a dubious claim"

A.4 Myths: metaphors and narratives (longer term change)

Lastly, there are **myths or metaphors** (e.g., Alexander's idea that the architect's work is done 'for the glory of God' (see Galle [2020]) or his conception that 'primitive' dwellings contain more life). To emphasize, CLA does not dismiss myths in the slightest: on the contrary, they are what drive the other layers. Another term that is used to characterize this layer is **narratives**. VanDrunen surfaced various concepts in Gabriel's essay that would be at home at this level, for example, the concept of duende that Gabriel takes over from Garcia Lorca originally derives from *dueño de casa*, the name of a certain kind of household spirit. However, myth here does not just refer to such entities, but to the most deeply held beliefs and concepts that underlie worldviews.

One important narrative for the pattern discourse is in plain view within the terminology of problems and solutions, which come from mathematics and physics. Alexander worked *at the level of narrative* to connect the patterns discourse to a scientific worldview, seeking a sense of objectivity. For example, in "The Atoms of Environmental Structure":

> most designers ... say that the environment cannot be right or wrong in any objective sense but that it can only be judged according to criteria, or goals, or policies, or values, which have themselves been arbitrarily chosen. We believe this point of view is mistaken.

Notice that, here, the discourse is positioned as different from the mainstream of architecture. The key differentiator is not the language of problems and solutions, which would be familiar to anyone with an engineering background; rather, but in a certain notion of *wholeness*. Which notion of wholeness remains to be surfaced. Quoting, again, from "The Atoms of Environmental Structure", we get some relevant background:

> We believe that all values can be replaced by one basic value: everything desirable in life can be described in terms of freedom of people's underlying tendencies. . . . The environment should give free rein to all tendencies; conflicts between people's tendencies must be eliminated.

Historically, there are at least two major varieties of wholeness: one that is based on progressive differentiation (e.g., unfolding from substance, per Spinoza), and the other generated by interaction between components (e.g., mutually reflecting monads, per Leibniz). In support of these allusions, a quote of Alexander from *The Nature of Order* (*TNO*): it "may be best if we redefine the concept of God in a way that is more directly linked to the concept of 'the whole.'" Indeed, the pattern discourse appears to draw from *both* major traditions of wholeness, while also seeking to unite them. We get the idea of unfolding in *APL* and other pattern languages that work in a top-down manner: however, we also get the notion of patterns and principles that are generative of emergent phenomena.

At this level, architecture and programming were seen, by Alexander [1999], to unite: his questions for the computer scientists to whom he was speaking point in the direction of bio-hacking and nanotechnology (e.g., for molecular self-assembly)—at least at the allusive level. The following quote suggests we have embarked on a fruitful track by attempting to think at the deeper layers of the pattern discourse:

> Generative patterns work indirectly; they work on the underlying structure of a problem (which may not be manifest in the problem) rather than attacking the problem directly.

The prominence of linguistic metaphors within DPL reminds us that Alexander's architectural oeuvre contains many traces of symbols associated with Hermes: a deity associated with communication and mediation. Through these reflections we gain some useful DISTANCE.

> In the house, [Hermes'] place is at the door, protecting the threshold. . . He could be found around city gates, intersections, state borders, and tombs (the gateways to the other world). [Benvenuto 1993]

At the time when Hermes was actively embraced as a deity, in some traditions he was paired with Hestia, the goddess of the hearth, whose "domain was internal, the closed, the fixed, the inward" (*ibid.*, here and in quotes later in this paragraph). The discourse around patterns contains some aspects that move towards foundations (e.g., in the form of fundamental principles, per *TNO*). Such foundations could be associated with Hestia, whereas Hermes would be on the side of generativity and mutation. The dichotomy seems to repeat itself within the *TNO* principles themselves: recalling that "focus" is the Latin term for the hearth, Strong Centers would align with Hestia, whereas Hermes would align more with Deep Interlock and Ambiguity. The resolution of the two forces within pattern language—as a form—seems be a variation of these Nietzschean lines: "anything that is becoming returns" (i.e., is discussable as pattern), and, "contingency resolves itself into necessity" (i.e., the wholeness of generativity ultimately recovers the wholeness of unfolding).

B. SUPPLEMENT: CASE STUDY: PLANNING "SEASON 1" FOR THE EMACS RESEARCH GROUP

In the Emacs Research Group, we did a PAR at the end of every session, in our (approximately weekly, two-hour) meetings from November 2020 to the time of writing. This allowed us to track progress, and to surface key issues and concerns. For example, bootstrapping needs related to scheduling and collaboration tools, along with persistent questions about how best to go public, are documented in our first PAR. Every six weeks or so, we merged selected bullet-points from the collected PARs into the CLA outline in an intuitive way, depending on which section they seemed to fit best. We elaborated those bullet points into a narrative form, which we jointly revised to accommodate new data as time went on. We also began to develop TODO items that would make the next steps for this seminar group both actionable and meaningful. Additionally, we connected these TODO items to design patterns collected in the *Peeragogy Handbook* [Corneli et al. 2016] (with ongoing work appearing at `peeragogy.org`). The TODO items typically are not concrete objectives, but are, rather, descriptions of anticipated patterns of behavior—here linked to *bona fide* design patterns. Some new proto-patterns are named in all-caps. To

refine these items into tasks that are concretely doable will require further breakdown, refinement, and elaboration; furthermore, managing the TODO list as a whole will require ongoing (re-)prioritization. Typically, TODO items at the Litany level are more immediately actionable, whereas those at the lower levels may take longer to realise.

Paragraphs summarising the CLA are augmented with representative data from the seminar sessions, and further broken down into next steps which are cross-referenced with peeragogy design patterns, like ROADMAP [Corneli et al. 2016]. By the time of our fourth iteration of the larger $\mathrm{PAR} \to \mathrm{CLA}$ cycle, each section had accumulated around 20-30 bullet points of supporting data at a similar level of granularity.

B.1 Understanding data, headlines, empirical world (short term change)

We've made progress since we started with the raw themes of Research on/in/with Emacs back in November 2020. We've met almost every week since then, and interviewed some interesting and varied guests. We have a clearer idea of what we want to talk about at the next EmacsConf, and how we can be of service to researchers and Emacs users. We have been using a workflow that helps us carefully review progress, diagnose issues, and manage our energy. We're understanding how research is done by doing it, and keeping careful track of the process. If a session doesn't go as well as hoped, we think about why (especially the chair). The idea is that you know what the bomb is, so you can at least hope to defuse it later. We try to adapt gracefully to circumstances as they evolve, without being reactive because we know we will be back again next week and the week after, etc.

Representative supporting data

(1) Everyone shared a brief intro and ideas so we got to know each other

(10) We've brainstormed a couple of options for getting out there: White-papers, Grants, Journal papers (very concrete)

(18) Alex: My major intention was to meet you guys and learn something, wanting to reinforce existing knowledge of emacs and develop it further

Next Steps

Maintain plans for the next six months	ROADMAP
Process the following points	SCRAPBOOK
Keep doing PARs and CLAs	ASSESSMENT
Develop our intention-based workflow	FORUM
Mesh with other ongoing activities elsewhere	COOPERATION

B.2 Systemic approaches and solutions (social system)

If we tackle big enough projects, it will bring with it the need for collaboration. (And we need to respect these other parties.) We like to create tangible deliverables (e.g., journal articles). However, "If we knew what the outcome was it wouldn't be research" — therefore, we're focusing initially on research methods and design documents. So far many of the stakeholders have to do with free software, open communities, peer learning: all of this is part of a broader initiative. All of them will need some degree of systematised activities and documents. This is what we're experimenting with; in principle, we can provide meta-management workflows. In the way we work together we make sure to take account of emotions, not just a time table. We've been experimenting with futures methods that help us use the future intelligently: neither trying to do everything 'live', nor overburdening the future with a bunch of plans that can't be realised.

Representative supporting data

(1) Part of a greater sense of trying to do something with EmacsConf to federate the community

(1) Joe: Leo did an amazing job facilitating the meeting

(1) Public Policy conference: (How to get a grant?)

(10) Potential interview with Leo & Jethro Kuan (co-maintainers of org-roam)

(18) Leo did a nice job of intervening

Next Steps

Identify potential stakeholders in Emacs Research	COMMUNITY
Identify stakeholders in the kind of activities we can support	A SPECIFIC PROJECT
Identify venues where we can reach these different stakeholders	WRAPPER
Create some publication to plant a flag for our group	PAPER
Keep exploring!	SERENDIPITY

B.3 Worldview, ways of knowing and alternative discourse

We have looked at RStudio and Roam Research as models of (some of) the kinds of things we think Emacs can eventually improve upon. Actually getting there requires thinking about the specificity of what Emacs can do. At least nominally, it is a system for editing. For example, currently we can edit a wiki using Org Roam and Git. What about editing distributed knowledge graphs? This would allow people to reference ongoing research processes. Or we could go further and contribute to the development of a new distributed read-write Web! Alongside such software products could come various services, such as a matchmaking service for academics, or a set of 24/7 virtual conferences. To make such a thing really useful, we need to get coherence out of various long-running, diverse, and heterogeneous thought processes: and make sure we help people address real problems. We can start small, working with the members of ERG and their networks. We're certainly not the only people who are struggling with some unexpected commitments. Successful adaptation requires the articulation of an entire system. Thinking again about Emacs: its current documentation is certainly useful, but it leaves many gaps, some of which are filled in other ways (e.g., by mailing lists). As we work we are paying attention to the growth not only of knowledge, but also of capability. For this, we often rely on our feelings (getting it, not getting it, accomplishing something or not, etc.).

Representative supporting data

(1) Wonderful outcome from attending EmacsConf 2020!

(10) Anthropology + Psychology is a special nightmare for reproducibility

(10) Maybe the ERG could contribute further patterns?

(18) But there's a problem with Emacs, which is that there isn't proper intro

Next Steps

Survey related work	CONTEXT
Spec out the Emacs based 'answer' to RStudio, Roam Research	COMMUNITY
Continue to develop and refine our methods	ASSESSMENT
Product and business development plans for a multigraph interlinking service	WEBSITE

B.4 Myths, metaphors and narratives: imagined (longer term change)

In our concrete methods, we have aligned ourselves with the 'long-term perspective'. This includes both retrospective and prospective thinking. For example, the things that were timely 7 years ago might not be now; in many cases the relevance of a given innovation goes down over time. That said, Emacs has an evolutionary character that has allowed it to keep up with the times — apparently becoming increasingly relevant and useful ever since Steele and Stallman started to systematise Editor MACroS for the Text Editor and Corrector (TECO) program. Not only has the technology evolved, but so has the social setting in which this work is done. After nearly a year of working together, we're now prepared to argue that other people should consider getting together to create their own communities similar to ours! The concepts underlying the free software movement were based on

"communal sharing" of source code: now we're working on developing and sharing other methods as well. Just like free software has legitimately expanded the range of what's humanly possible, so too may further efforts expand the frontier of possibility. As above, this takes careful articulation, and a willingness to do truly original research (without relying on fixed assumptions about what research is meant to be, or where we will find value).

Representative supporting data

(1) We generally agreed that we want to make something that exposes intrinsic value of using these tools

(10) [None recorded at this level from this PAR.]

(18) But there was no such guidance; you were in the middle of an alien playground. "But I just wanted to do my Clojure stuff."

Next Steps

Assess what we're learning	ASSESSMENT
Think about how we can help improve gender balance in Free Software	DIVERSITY

REFERENCES

Christopher Alexander. 1964. *Notes on the Synthesis of Form*. Harvard University Press.

C. Alexander. 1979. *The Timeless Way of Building*. Oxford University Press.

C. Alexander. 1999. The origins of pattern theory: the future of the theory, and the generation of a living world. *IEEE Software* 16, 5 (1999), 71–82. DOI:http://dx.doi.org/10.1109/52.795104

Robert Arnkil. 2008. Remembering the future: Future dialogue and the future of dialogising. In *Dialogue in Working Life Research and Development in Finland*. Peter Lang, Frankfurt am Main, 131–143.

Stephen Batchelor. 2020. Embracing Extinction. *Tricycle* (Fall 2020). https://tricycle.org/magazine/stephen-batchelor-climate/ Accessed 23 August, 2021.

Gregory Bateson. 1979. *Mind and nature: A necessary unity*. Bantam Books New York.

Sergio Benvenuto. 1993. Hermes-Hestia: The Hearth and the Angel as a Philosophical Paradigm. *Telos* 1993, 96 (1993), 101–118.

Jan O Borchers. 2008. A pattern approach to interaction design. In *Cognition, Communication and Interaction*. Springer, 114–131.

Greg Bryant. 2015. Living Structure and Cognition. *RAIN Magazine* (25 June 2015). http://www.rainmagazine.com/archive/2015/living-structure-and-cognition Accessed 24 November, 2020.

J. Campbell. 1949. *The Hero With A Thousand Faces*. Pantheon Books (The Bollingen Series).

Russell Clemens. 2020. Asian Elephant Futures: A Causal Layered Analysis with Gregory Bateson in Mind. *World Futures Review* 12, 1 (2020), 55–80.

Joseph Corneli. 2014. Peer Produced Peer Learning: A Mathematics Case Study. (2014). http://oro.open.ac.uk/40775/

Joseph Corneli, Charles Jeffrey Danoff, Charlotte Pierce, Paola Ricaurte, and Lisa Snow MacDonald (Eds.). 2016. *The Peeragogy Handbook: The No-Longer-Missing Guide to Peer Learning & Peer Production* (3 ed.). PubDomEd/Pierce Press. 272 pages. http://peeragogy.org

Joseph Corneli, Simon Holland, Alison Pease, Paul Mulholland, Dave Murray-Rust, Theodore Scaltsas, and Alan Smaill. 2018. Patterns of Design. In *Proceedings of the 23rd European Conference on Pattern Languages of Programs*. ACM. DOI:http://dx.doi.org/10.1145/3282308.3282331

W. Cunningham and M.W. Mehaffy. 2013. Wiki as pattern language. In *Preprints of the 20th Pattern Languages of Programs Conference*, Vol. PLoP'13. Oslo Norway, 32. http://dl.acm.org/citation.cfm?id=2725669.2725707 Accessed 24 November, 2020.

Marilyn Darling, Charles Parry, and Joseph Moore. 2005. Learning in the Thick of It. *Harvard Business Review* (July-August 2005). https://hbr.org/2005/07/learning-in-the-thick-of-it Accessed 24 November, 2020.

M.J. Dawes and M.J. Ostwald. 2017. Christopher Alexander's A Pattern Language: analysing, mapping and classifying the critical response. *City, Territory and Architecture* 4, 1 (2017), 1–14. DOI:http://dx.doi.org/10.1186/s40410-017-0073-1

Michael J Dawes and Michael J Ostwald. 2018. The mathematical structure of Alexander's A Pattern Language: An analysis of the role of invariant patterns. *Environment and Planning B: Urban Analytics and City Science* 47, 1 (Feb. 2018), 7–24. DOI:http://dx.doi.org/10.1177/2399808318761396

Pedro Fidelman, Truong Van Tuyen, Kim Nong, and Melissa Nursey-Bray. 2017. The institutions-adaptive capacity nexus: insights from coastal resources co-management in Cambodia and Vietnam. *Environmental Science & Policy* 76 (2017), 103–112.

Design Friction. 2016. Flaws of the Smart City [Card game]. (2016). http://www.flawsofthesmartcity.com/ Accessed 25 November, 2020.

Karl Friston. 2013. Life as we know it. *Journal of The Royal Society Interface* 10, 86 (Sept. 2013), 20130475. DOI:http://dx.doi.org/10.1098/rsif.2013.0475

R.P. Gabriel. 2002. *Writers' Workshops & the Work of Making Things: Patterns, Poetry...*. Pearson Education.

Richard P Gabriel and Ron Goldman. 2000. Mob software: The erotic life of code. In *Proc. ACM Conf. Object-Oriented Programming, Systems, Languages, and Applications*.

Per Galle. 2020. Christopher Alexander's Battle for Beauty in a World Turning Ugly: The Inception of a Science of Architecture? *She Ji: The Journal of Design, Economics, and Innovation* 6, 3 (2020), 345–375. DOI:http://dx.doi.org/https://doi.org/10.1016/j.sheji.2020.03.002

D. Handelman. 1998. *Models and mirrors: Towards an anthropology of public events*. Berghahn Books.

Sirkka Heinonen, Matti Minkkinen, Joni Karjalainen, and Sohail Inayatullah. 2017. Testing transformative energy scenarios through causal layered analysis gaming. *Technological Forecasting and Social Change* 124 (2017), 101–113. DOI:http://dx.doi.org/https://doi.org/10.1016/j.techfore.2016.10.011

Ernest R. Hilgard. 1980. The trilogy of mind: Cognition, affection, and conation. *Journal of the History of the Behavioral Sciences* 16, 2 (April 1980), 107–117. DOI:http://dx.doi.org/10.1002/1520-6696(198004)16:2<107::aid-jhbs2300160202>3.0.co;2-y

Dave Hoover and Adewale Oshineye. 2009. *Apprenticeship patterns: Guidance for the aspiring software craftsman*. O'Reilly Media, Inc.

Lewis Hyde. 2019. *The gift: How the creative spirit transforms the world*. Vintage.

S. Inayatullah. 1998. Poststructuralism as method. *Futures* 30, 8 (1998), 815–829. DOI:http://dx.doi.org/10.1016/S0016-3287(98)00086-X

Sohail Inayatullah. 1999. *Situating Sarkar: Tantra, Macrohistory and Alternative Futures*. Gurukul Publications.

Sohail Inayatullah. 2004. *The Causal Layered Analysis (CLA) Reader: Theory and Case Studies of an Integrative and Transformative Methodology*. Tamkang University Press.

Paul Salvador Inventado and Peter Scupelli. 2017. Towards a Community-Centric Pattern Repository. In *Proceedings of the 22nd European Conference on Pattern Languages of Programs (EuroPLoP '17)*. Association for Computing Machinery, New York, NY, USA, Article 36, 4 pages. DOI:http://dx.doi.org/10.1145/3147704.3147743

Asghar Iran-Nejad and Adam Winsler. 2000. Bartlett's schema theory and modern accounts of learning and remembering. *The Journal of Mind and Behavior* (2000), 5–35.

F. Jacobus. 2009. The Disappearing Architect: 21st Century Practice and the Rise of Intelligent Machines. In *97th ACSA Annual Meeting Proceedings, The Value of Design*. 159–165. https://www.acsa-arch.org/proceedings/ Accessed 24 November, 2020.

C. Kohls. 2010. The structure of patterns. In *Proceedings of the 17th Conference on Pattern Languages of Programs - PLOP*, Vol. 10. 1–10. DOI:http://dx.doi.org/10.1145/2493288.2493300

Christian Köppe, Paul Salvador Inventado, Peter Scupelli, and Uwe Van Heesch. 2016. Towards Extending Online Pattern Repositories: Supporting the Design Pattern Lifecycle. In *Proceedings of the 23rd Conference on Pattern Languages of Programs (PLoP '16)*. The Hillside Group, USA, Article 15, 26 pages.

Aaron Krowne and Raymond Puzio. 2006. The fog of copyleft. *First Monday* 11, 7 (Jul. 2006). DOI:http://dx.doi.org/10.5210/fm.v11i7.1368

Aaron Phillip Krowne. 2003. An Architecture for Collaborative Math and Science Digital Libraries. (2003). https://vtechworks.lib.vt.edu/handle/10919/34891

H. Leitner. 2015. *Pattern theory: introduction and perspectives on the tracks of Christopher Alexander*. HLS Software.

Alexandre Magnan. 2010. *For a better understanding of adaptive capacity to climate change: a research framework*. Technical Report. Institut du developpement durable et des relations internationales-IDDRI.

Grégoire Mallard. 2019. *Gift Exchange*. Cambridge University Press. DOI:http://dx.doi.org/10.1017/9781108570497

MW Mehaffy, Yulia Kryazheva, Andrew Rudd, Nikos A Salingaros, and others. 2020. A New Pattern Language for Growing Regions. (2020).

Riel Miller. 2018. *Transforming the future: Anticipation in the 21st century*. Taylor & Francis.

T.P. Moran. 1971. (Artificial, intelligent) architecture: Computers in design. *Architectural Record* 149 (1971), 129–134.

Allen Newell. 1983. The heuristic of George Polya and its relation to artificial intelligence. *Methods of heuristics* (1983), 195–243.

Friedrich Nietzsche. 2017. *Nietzsche: on the genealogy of morality and other writings*. Cambridge University Press.

J. Nobel and R. Biddle. 2002. Notes on postmodern programming. In *Proceedings of the Onward Track at Oopsla 02, the ACM conference on Object Oriented Programming, Systems, Languages and Applications*, R Gabriel (Ed.). Seattle, USA, 49–71.

L. K. S. Olabisi, A. R. Kapuscinski, K. A. Johnson, P. B. Reich, B. Stenquist, and K. J. Draeger. 2010. Using scenario visioning and participatory system dynamics modeling to investigate the future: Lessons from Minnesota 2050. *Sustainability* 2, 8 (2010), 2686–2706. https://doi.org/10.3390/su2082686

E. Ostrom. 2009. *Understanding Institutional Diversity*. Princeton University Press. https://doi.org/10.2307/j.ctt7s7wm

Rivka E Oxman. 1994. Precedents in design: a computational model for the organization of precedent knowledge. *Design Studies* 15, 2 (1994), 141–157. DOI:http://dx.doi.org/https://doi.org/10.1016/0142-694X(94)90021-3

George Polya. 1945. *How to solve it: A new aspect of mathematical method*. Princeton University Press.

George Polya. 1954. *Mathematics and Plausible Reasoning: Volume II, Patterns of Plausible Inference*. Princeton University Press.

Douglas Rice, Saba Siddiki, Seth Frey, Jay H. Kwon, and Adam Sawyer. 2021. Machine coding of policy texts with the Institutional Grammar. *Public Administration* 99, 2 (Jan. 2021), 248–262. DOI:http://dx.doi.org/10.1111/padm.12711

Ingrid Robeyns and Morten Fibieger Byskov. 2021. The Capability Approach. In *The Stanford Encyclopedia of Philosophy* (Winter 2021 ed.), Edward N. Zalta (Ed.). Metaphysics Research Lab, Stanford University.

Geir Kjetil Sandve, Anton Nekrutenko, James Taylor, and Eivind Hovig. 2013. Ten simple rules for reproducible computational research. *PLoS Comput Biol* 9, 10 (2013), e1003285.

Marco Schorlemmer and Enric Plaza. 2021. A uniform model of computational conceptual blending. *Cognitive Systems Research* 65 (2021), 118–137. DOI:http://dx.doi.org/https://doi.org/10.1016/j.cogsys.2020.10.003

P. Schwartz. 1996. *The Art of the Long View: Paths to Strategic Insight for Yourself and Your Company*. Doubleday.

A. Shalloway and J. Trott. 2005. *Design patterns explained: A new perspective on object-oriented design*. Addison-Wesley Professional.

Herbert A. Simon and Allen Newell. 1958. Heuristic Problem Solving: The Next Advance in Operations Research. *Operations Research* 6, 1 (1958), 1–10. http://www.jstor.org/stable/167397

Richard A. Slaughter. 1989. Probing beneath the surface: Review of a decade's futures work. *Futures* 21, 5 (1989), 447–465. DOI:http://dx.doi.org/https://doi.org/10.1016/0016-3287(89)90085-2

J. Sutherland. 2019. *A Scrum Book: The Spirit of the Game*. The Pragmatic Programmers.

Toufik Taibi and David Chek Ling Ngo. 2003. Formal Specification of Design Patterns - A Balanced Approach. *Journal of Object Technology* 2, 4 (2003), 127–140.

Christoph Teufel and Paul C. Fletcher. 2020. Forms of prediction in the nervous system. *Nature Reviews Neuroscience* 21, 4 (01 Apr 2020), 231–242. DOI:http://dx.doi.org/10.1038/s41583-020-0275-5

Le Thi Hong Phuong, G. Robbert Biesbroek, and Arjen E.J. Wals. 2017. The interplay between social learning and adaptive capacity in climate change adaptation: A systematic review. *NJAS - Wageningen Journal of Life Sciences* 82 (2017), 1–9. DOI:http://dx.doi.org/https://doi.org/10.1016/j.njas.2017.05.001

Kirsten Thonicke, Michael Bahn, Sandra Lavorel, Richard D Bardgett, Karlheinz Erb, Mariasilvia Giamberini, Markus Reichstein, Björn Vollan, and Anja Rammig. 2020. Advancing the understanding of adaptive capacity of social-ecological systems to absorb climate extremes. *Earth's Future* 8, 2 (2020), e2019EF001221.

Jenifer Tidwell. 2010. *Designing interfaces: Patterns for effective interaction design*. O'Reilly Media, Inc.

Cameron Tonkinwise and Abby Mellick Lopes. 2021. How new design patterns can enable cities and their residents to change with climate change. *The Conversation* (16 February 2021). https://theconversation.com/how-new-design-patterns-can-enable-cities-and-their-residents-to-change-with-climate-change-152749 Accessed 14 June, 2021.

Roberto Mangabeira Unger. 2019. *The knowledge economy*. Verso Books.

US Army. 2002. Training the Force. (October 2002).

Thomas VanDrunen. 2010. A Christian Analysis of Gabriel's "Mob Software". https://acmsonline.org/home2/wp-content/uploads/2016/05/VanDrunen2010.pdf, (2010).

D. Wildschut. 2017. The need for citizen science in the transition to a sustainable peer-to-peer-society. *Futures* 91 (2017), 46–52.

PLoP'21, OCTOBER 4–7, Virtual Online. Copyright 2021 is held by the author(s). HILLSIDE 978-1-941652-17-6