D5.1
Computational account of serendipity

Authors	Joseph Corneli, Alison Pease, Simon Colton
Reviewers	Tarek Besold

Grant agreement no.	611553
Project acronym	COINVENT - Concept Invention Theory
Date	September 24, 2014
Distribution	PU/RE/CO

Disclaimer

The project COINVENT acknowledges the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open Grant number 611553.

Abstract

Drawing on well-known examples in scientific discovery, we develop a set of criteria for serendipity that can be applied as evaluation standards in computational settings. We draw on the notion of design patterns as a way to describe structure and order behaviour, emphasising the organic growth of a pattern language as a way to think about the discovery *and* invention that comprise serendipitous encounters. We show how several earlier patterns of serendipity could be applied in a Writers Workshop for computational systems, and include related recommendations for practitioners.

Keyword list: **serendipity**, **design patterns**, **intelligent machinery**, **Writers Workshops**.

Changes

Version	Date	Author	Changes
0.1	11.09.14	Joseph Corneli	Creation
0.2	13.09.14	Joseph Corneli	Abstract and executive summary
0.3	24.09.14	Joseph Corneli	Address comments from Tarek
0.4	29.09.14	Joseph Corneli	Text and comments from Simon

Executive Summary

Serendipitous discovery is related, firstly, to deviations from familiar patterns, and secondly, to new insight. It is the unexpected that is found to be both explicable and useful that we call serendipitous.

We briefly list some of the most famous examples of serendipitous discovery, referring to other book-length collections where further details can be found. We use these central examples to build a description of the key criteria, dimensions, and contributing factors of serendipitous discovery. The central criterion is a Focus Shift, corresponding to the discovery of something unexpected, together with the invention of an application for the same. The four key components that implement the focus shift are a Prepared Mind, a Serendipity Trigger, a Bridge, and a Result. We describe four additional situational dimensions that are generally present, and four supporting environmental factors, and note that serendipity can apply both on the individual and social scale. We then use these 13 dimensions to analyse an earlier collection of examples from Pek van Andel, extracting four "perfect" examples of serendipity that match all of our criteria.

Since serendipity features both discovery and invention, we argue that it is best approached through the growth and development of patterns in response to encounters with phenomena outside of the direct control. We point to Hofstadter and Mitchell's COPYCAT system as an exemplar of such an "emergent" computational architecture. We use our core criteria of the Focus Shift, Prepared Mind, Serendipity Trigger, Bridge, and Result to describe criteria for evaluating serendipity in a computational setting, and walk through an extended example describing a (hypothetical) Writers Workshop for Systems. We show how the four "perfect" examples of serendipity that we identified could be applied in this setting to the benefit of participating systems.

We briefly summarise related work, and offer our recommendations for practitioners in the field of computational creativity. In sum, these are:

(1) Progress should be evaluated in terms of the problems that system solve, *and* the degree to which the computer was responsible for coming up with this problem.

(2) Computationally creative computer systems should be directly involved in the evaluation of other computationally creative computer systems.

In concluding, we point out that serendipity, featuring both discovery that is not pre-planned and the invention of useful applications, should itself be used as a key criterion for computational creativity.

Contents

1 Introduction

Materials, artefacts, and processes have no value without a context of application. In practise, we are likely to attribute value to materials that are *useful*, and to attribute *creativity* to a person who puts materials to good use. Many instances of serendipity centre on reevaluation or reassessment. For example, a non-sticky "superglue" that no one was quite sure how to use turned out to be just the right ingredient for 3M's Post-it™ notes. Serendipitous discovery is related, firstly, to deviations from familiar patterns, and secondly, to new insight. *It is the unexpected that is found to be both explicable and useful that we call serendipitous.* When we consider the practical uses for weak glue, the possibility that a life-saving antibiotic might be found growing on dirty petri dishes, and or the idea that cockle-burs could be anything but annoying, we encounter radical changes in the evaluation of what's interesting.

The structure of the current paper follows the general outlines of an earlier survey from Pease et al. [52], but goes beyond it by presenting a unified theoretical perspective on serendipity, and a correspondingly strong stance on why robust serendipity-enhancing practises are important for progress in computational creativity. Section 2 develops 13 key criteria for the evaluation of serendipity based on a review of several well-known examples of serendipitous discoveries from human history. In Section 4 we apply these criteria to analyse several narrative "patterns of serendipity" that have been collected by Pek van Andel [64], and develop a general-purpose approach to understanding serendipity, drawing on the literature from the design pattern community. In Section 5, we focus in on serendipity in a computational context, condensing our criteria into an operational definition, and propose an experimental setup that we think will exhibit many of the relevant features. In Section 6, we examine related work, and in Section 7, we advance our recommendations for researchers working on computational creativity (and serendipity).

Van Andel – echoing Poincaré's [53] (negative) reflections on the potential for a purely computational approach to mathematics – claimed that:

> "*Like all intuitive operating, pure serendipity is not amenable to generation by a computer. The very moment I can plan or programme 'serendipity' it cannot be called serendipity anymore*." [64]

We believe that serendipity is not so mystical as such statements might imply, and in Sections 4 and 5 we show how it is possible to reinterpret van Andel's patterns of serendipity for computational settings. However, we do not propose to "programme serendipity," but rather to programme with serendipity in mind. In an essay from the late 1960's, Minsky [49] takes up a theme for which he would become well known, and suggests that in practise, programmers write programs "*for the individuals of little societies*" – precisely because we cannot envision in advance all of the details of program interactions. The real problem with computers is *not* that they only do what they're told. Indeed, indeterminacy forms an important part of any proposal for "intelligent machines", after Turing [63]:

> "*They will make mistakes at times, and at times they may make new and very interesting statements, and on the whole the output of them will be worth attention to the same sort of extent as the output of a human mind*."

The role of serendipity in computer science is perhaps foundational in several respects: from the point of view of the large scale history of the field [19], and from the point of the micro-history of every computational discovery. We aim to significantly clarify the latter aspect here.

2 Literature review

2.1 Serendipity in a human context

Serendipity is a value-laden concept, and has variously been considered to depreciate and to enhance a scientist's achievement. This has lead to accounts in which the role of serendipity in a discovery is either under- or over-rated. Despite this, the related notions of luck, skill, and happy accident make for exciting reading, and numerous examples have been gathered into collections. Merton and Barber [48] trace the history of the word "serendipity" from its coinage by Horace Walpole in 1754 up to 1954 (with an extended afterword on its usage from 1954–2004); [57] contains over 70 examples; [59] contains examples in cosmology, astronomy, physics and other domains; and in [64], van Andel claims to have documented over 1000 examples, some of which appear in [65]. Umberto Eco [27] offers another survey, focusing particularly on serendipitous mistakes. As detailed descriptions can be found in other places, we briefly list some of the most famous examples.

- The 17th Century discovery that *quinine* extracted from the bark of South American cinchona trees could be used to treat and prevent malaria – building on a much earlier indigenous Quechua discovery that the extract stops shivering.
- Fleming's discovery of *penicillin*.[1]
- de Mestral's invention of *Velcro*™ following the model presented by cockle-burs that stuck to his jacket while out walking [57, pp 220-222].
- Arthur Fry's invention of sticky bookmarks (the prototype for *Post-it*™ notes), using a weak glue developed by his colleague, Spencer Silver [57, p. 224].
- Penzias and Wilson's discovery of the *echoes of the Big Bang* [62].
- Kekulé's dream-inspired discovery of the *structure of the benzine ring* [5, p. 21], cf. [57, p. 77].
- Charles Goodyear's invention of *vulcanised rubber* [35].
- The *Rosetta Stone* was found by a soldier who was demolishing a wall in order to clear ground for what was to be Fort St. Julien [57, pp. 109 - 111].

It is not hard to spot further examples with similar characteristics: e.g. the invention of dry cleaning by a professional dye-maker after his maid spilled kerosene on the tablecloth, or the discovery of a marketable use for sildenafil citrate (better known as *Viagra*™) which had been trialled as a heart medicine. Importantly, there are examples that exhibit features of serendipity that develop on a social scale, rather than an individual scale. Between Spencer Silver's creation of high-tack, low-adhesion glue in 1968, the invention of a sticky bookmark in 1973, and the eventual launch of the distinctive canary yellow re-stickable notes in 1980, there were many opportunities for Post-its™ *not* to have come to be [29]. Large-scale scientific and technical projects – like the 17th-18th Century project to map China – generally rely on the "*convergence of interests of several key actors*" [13], along with other supporting cultural factors.

[1] Merton and Barber [48] state that the description of this discovery was the first time that the word *serendipity* was used without inverted commas or accompanying definition.

2.2 Definitions of serendipity

Merton and Barber [48] highlight three points of particular interest to us.

(1) While Walpole unambiguously insisted that serendipity referred to an *unsought* finding – "***No** discovery of a thing you **are** looking for comes under this description*" – this criterion has dropped from dictionary definitions. In 30 English language dictionaries from 1909 - 2000, only five explicitly say "not sought for" (see [57, pp. 246–249] for a table of the definitions). Certainly, some of our examples of serendipity are of *sought* findings (such as Goodyear's vulcanisation). Some writers classify such events as *pseudoserendipity* – a term coined by Roberts to describe accidental discoveries of ways to achieve an end which is sought for [57, p. x].

(2) While Walpole initially described serendipity as an event (a discovery), it has since been reconceptualised as a psychological attribute, a matter of sagacity on the part of the discovery: that is, a "gift" or "faculty" more than a "state of mind." In the list of 30 dictionary definitions, only one, from 1952, defined it solely as an event; and five defined it as both event and attribute.

(3) Merton and Barber argue that the psychological perspective needs to be integrated with a *sociological* one. That is, a personal characteristic like "sagacity" is not a full explanation:

> "*For if chance favours prepared minds, it particularly favors those at work in microenvironments that make for unanticipated sociocognitive interactions between those prepared minds. These may be described as serendipitous sociocognitive microenvironments*" [48, p. 259–260].

It may be relevant to note that serendipity is usually discussed within the context of *discovery* – rather than *creativity*, although in typical parlance these terms are closely related [40]. Henri Bergson's [7] distinction will be useful in what follows:

> "*Discovery, or uncovering, has to do with what already exists, actually or virtually; it was therefore certain to happen sooner or later. Invention gives being to what did not exist; it might never have happened.*"

In the field of computational creativity, the term "discovery" is often used in connection with scientific discovery, while the term "creativity" is applied to computationally similar processes in arts domains. However, in the choice of terminology and implied research focus, the relevance of the process of "discovery" may have been overlooked – while "creative" claims are not always warranted.

Serendipity, as we understand the term, would seem to require features of both, that is, the discovery of something unexpected, together with the invention of an application for the same. The balance between these two features will differ from case to case. In the following section, we will consider characteristics of serendipity with particular reference to the classic examples listed above.

2.3 Characteristics of serendipity

Here we will describe our key condition for serendipity, the presence of a Focus Shift, together with four key components that implement this (Prepared Mind, Serendipity Trigger, Bridge, Result), four dimensions that are generally present to some degree in instances of serendipitous discovery or invention (Chance, Curiosity, Sagacity, Value) and four supporting environmental factors that are, if not strictly required, at least conducive to serendipity (Dynamic world, Multiple contexts, Multiple tasks, Multiple influences).

Key condition for serendipity

Focus shift. The most extreme cases show focus establishing itself as if from nowhere: de Mestral was walking through the Alps when he encountered the "seeds" of his discovery. In some cases, the focus shift takes place within a social context: Arthur Fry and Spencer Silver had different ideas about what could be done with weak glue. In all of the discoveries listed in Section 2.1, there was a radical change in the discoverer's evaluation of what is interesting. We can think of this as a reclassification of "noise" to "signal."

Components of serendipity

Prepared Mind. Kekulé's "prepared mind" included his focus on the problem of finding the structure of the benzine molecule and his knowledge and skill as a scientist. Fleming's "prepared mind" included his focus on carrying out experiments to investigate influenza as well as his previous experience that foreign substances in petri dishes can kill bacteria. He was concerned above all with the question "*Is there a substance which is harmful to harmful bacteria but harmless to human tissue?*" [57, p. 161]. The social analogues are clear: for example, 3M not only had a talented staff, but ran internal technical forums where staff members could exchange ideas.

Serendipity Trigger. The trigger does not directly cause the outcome, but rather, inspires thought. Indeed, the trigger may bear very little resemblance to the eventual result. On its own, the trigger would not typically be seen as an important discovery. Examples include a dream, a petri dish with a clear area, and cockle-burs attached to a jacket. In a social context, the trigger may have several components, and may rely on the circumstantial alignment of interest between different parties.

Bridge. The bridge is what affords movement from the trigger to the result. These include reasoning techniques, such as abductive inference (what might cause a clear patch in a petri dish?); analogical reasoning (de Mestral constructed a target domain from the source domain of burs hooked onto fabric); and conceptual blending (Kekulé blended his knowledge of molecule structure with his vision of a snake biting its tail and invented the concept of a benzine ring). The bridge may also rely on new social arrangements, such as the formation of cross-cultural research networks [13].

Result. This is the outcome itself. This may be a new product, artefact, process, hypothesis, a new use for a material substance, and so on. The outcome may contribute evidence in support of a known hypothesis, or a solution to a known problem. Alternatively, the result may itself be a *new* hypothesis or problem. The result may be a "pseudoserendipitous" in the sense that it was *sought*, while nevertheless arising from an unknown, unlikely, coincidental or unexpected source. More classically, it is an *unsought* finding, such as the discovery of the Rosetta stone.

Dimensions of serendipity

Chance. The *serendipity trigger* tends to be unlikely, unexpected, unsought, accidental, random, or coincidental. The trigger has features that arise independently of the result, and even independently of any search for a result. The relevant features may be "hidden in plain view," and chance may apply to the conditions that eventuate in their discovery, as when malarial Europeans chanced upon a remedy found only in South America. Fleming [30] noted: "*There are thousands of different moulds*" – and "*that chance put the mould in the right spot at the right time was like winning the Irish sweep.*"

Curiosity. The capacity for keeping an *open mind*, and the corresponding ability to take advantage of the unpredictable, is necessary for a focus shift to take place. Many of the investigators described above went beyond simply keeping an open mind in order to actively exercise their curiosity about the way things work. Importantly, a preliminary evaluation of interestingness often takes place well before an final evaluation of the outcome. Venkatesh Rao [55] refers to a "cheap trick" that takes place early on in many narratives in order to establish preliminary conditions of order, and curiosity with respect to unexpected stimuli can play this role.

Sagacity. This old-fashioned word is related to "wisdom," "insight," and especially to "taste," and describes the attributes, or skill, of the discoverer that contribute to forming the bridge between the trigger and the result. In many cases, such as an entanglement with cockle-burs, many others will have already been in a similar position and not obtained an interesting result. Relevant skills include the ability to keep an open mind, to perform a focus shift, to see the value in a discovery, and to build a suitable bridge.

Value. It is generally agreed that a serendipitous result is one that is seen to be happy or useful. This judgement may be made independently (in the computational creativity context, Jordanous [40] argues that this is preferable) or, more commonly, by the discoverer/creator. Note that the chance "discovery" of, say, a £10 note may be seen as happy by the person who finds it, whereas the loss of the same note would generally be regarded as unhappy. Positive judgements of serendipity by a third party would be less likely in scenarios in which "One man's loss is another man's gain" than in scenarios where "One man's trash is another man's treasure."

Environmental factors

Dynamic world. Firstly, in the settings we are interested in, information about the world develops over time, and is not presented as a complete, consistent whole. Secondly, the components of serendipity as described above have an order of operations: the prepared mind takes the stage first, then the serendipity trigger takes place, a bridge is found, and after that the result. Value may come later. Van Andel estimates that twenty percent of innovations are based on "*something was discovered before there was a demand for it*" [64, p. 643].

Multiple contexts. One of the dynamical aspects at play may be the discoverer going back and forth between different contexts, with different stimuli. Thus, 3M employee Arthur Fry also sang in a church choir and needed a good way to mark pages in his hymn book. Malaria was not indigenous to Peru, where cinchona trees grow. Some contexts may play the role of a training ground for a subsequent discovery: for example, Goodyear had spent years experimenting with rubber using different processes before he hit upon the process of vulcanisation.

Multiple tasks. Even within what would typically be seen as a single context, a discoverer may take on multiple tasks that the context into sub-contexts, or that cause the investigator to look in more than one direction. Fleming happened to be doing the washing up after a holiday when he made his discovery. He might have overlooked the critical details had he not also been chatting with a former lab assistant who had stopped by. Penzias and Wilson were using a large antenna to detect radio waves that were relayed by bouncing off of satellites. Although they had removed effects due to radar, radio, and heat, they found residual ambient noise that couldn't be eliminated [67].

Multiple influences. A prepared mind, or its distributed analogue, may draw on a range of different skills and experiences. The "bridge" from trigger to result is often found through a social network, thus, for instance Penzias and Wilson only understood the significance of their work after reading a preprint by Jim Peebles that hypothesised the possibility of measuring radiation released by the big bang [67]. The process of discovery and invention may involve more than one "aha!" moment and skill set: Post-it™ notes again make a good example.

3 Foundational work

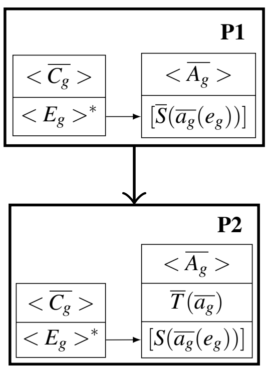


Figure 1: Progress in developing a poetry system

In [17], we introduced a diagrammatic formalism for keeping track of progress in creative computational systems. An example, pictured in Figure 1, shows how the second iteration of a poetry system becomes able to automatically apply aesthetic judgements in order to select the a preferred poem from a larger set of generated examples, once the programmer has translated (by hand) the relevant aesthetic measures. Progress, in this interpretation, amounts to more sophisticated processing, and, in particular, to the removal of "bars" – indicating that the system can do something that was formerly done by a programmer. Thus, this formalism keeps track both of the overall structure of computational systems, and which actors that are responsible for which actions within a given instance of the system.

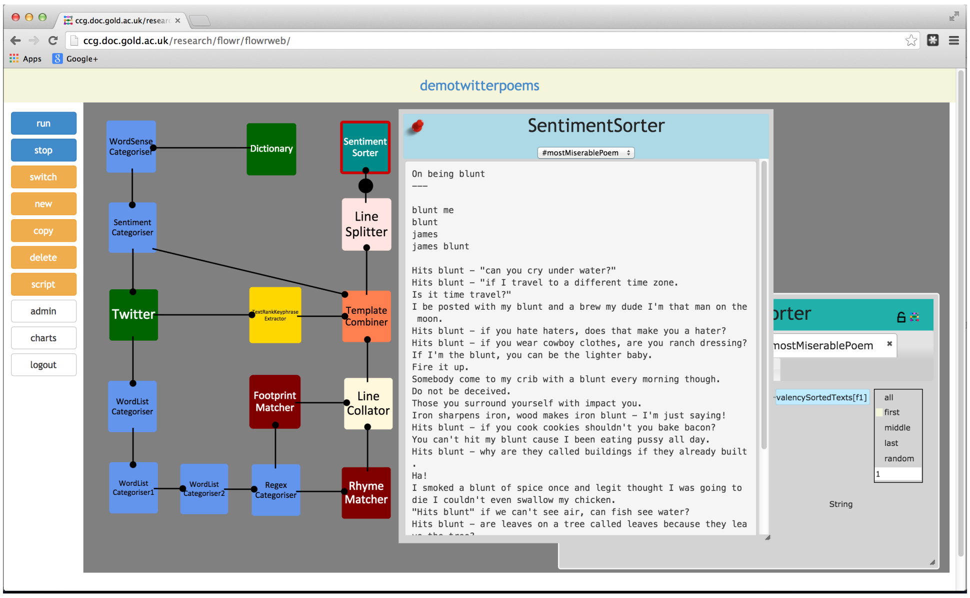


Figure 2: A sample poem generated by FloWr

As we develop models and systems that people would describe as serendipitous with reference to the criteria listed in in Section 2.3, there will be more to account for: for example, we will need to model dynamically changing environments and a computational version of a prepared mind.

To explore these aspects of the formalism, we have worked with a system called FloWr, which is portrayed in Figure 2. In FloWr, users can construct complex flowcharts composed of individual ProcessNodes, through which information flows and new information is generated. In the first part of the figure shown, a flowchart has constructed a poem based on live output from Twitter for the query "blunt". The dynamic aspect of this environment are threefold: (*i*) some of the nodes in the flowcharts access online news and social media sites, which change rapidly from minute to minute; (*ii*) the software itself can construct flowcharts, as described in [14], and shown in the background of the figure; and (*iii*) we are building a community of ProcessNode builders around the online version of FloWr, newly developed since the publication of [14], which we're building to help facilitate involvement of other Computational Creativity researchers.

When FloWr constructs flowcharts for itself, while each is semantically plausible (i.e., they pass the right type of data from ProcessNode to ProcessNode), many fail – for instance, because the available data is limited, or is narrowed down too quickly. In fact, the best results in [14] were at 20%, i.e., 80% of the flowcharts it constructed failed to produce output. Each of these failures can be saved as an outstanding problem in FloWr's prepared mind. As data changes and as new nodes are written and uploaded to the system, FloWr can replace nodes, update data, and in general change flowcharts in order to see if it could fix a broken flowchart.

For instance, suppose a ProcessNode developer wrote and uploaded a node to mine data from a new social network, in order, say, to produce text summarisations of world events. FloWr may take that node and substitute it in the place of an old "FaceBook" node in a broken poetry flowchart. If the replacement worked, and output was produced, this could be seen as a serendipitous occurrence: FloWr will have taken advantage of the dynamically changing environment – in which new social networks come and go, and in which text summaries may work better in some cases than in others – to resolve an outstanding problem in poetry generation.

The next stage for the FloWr system will be to modify it along these lines, to make it able to adapt to the dynamically changing environment, and to perform experiments where we monitor potentially serendipitous scenarios. Such experiments will be similar to those we tried with the HR2 system in [52], but improved because in this earlier effort, we had to break working processes in order to serendipitously fix them. The new experiments, the scenarios will be more realistic, i.e., there will be a catalogue of genuine open problems waiting to be solved. Understanding how to work with this catalogue and the associated experimental process will, of course, be used to further the computational model of serendipity. We discuss one direction for potential experimentation in Section 5.3.

4 Patterns of Serendipity

Figure 3 examines 14 situational patterns of serendipity collected in [64] through the lens of the evaluation criteria described in Section 2. As required by our theory, a "focus shift" appears in each instance, although it has a different flavour in the different examples. In this analysis, only three of the other criteria mentioned above are clearly present in *all* of the patterns: "a prepared mind", a "bridge", and a "dynamic world." Similarly, only four of Pek van Andel's patterns exhibit all of the characteristics we identified: *Successful error*, *Side effect*, *Wrong hypothesis*, and *Outsider*.

"Near misses" are also of interest, and help to understand the role of the factors from Section 2. For example, the *Inversion* pattern is somewhat closer to what is called an *antipattern* in the design pattern literature [10]. Van Andel describes the story of a researcher observing an effect (the anticoagulant heparine) which was precisely the opposite of the one sought (factors that *cause* blood clotting) – but failing to acknowledge that this observation was important for over 40 years. The result was eventually seen to be of value: however, in this instance, we may have an example of a mind that is *over-prepared* to find a particular sort of result, rather than a truly "sagacious" mind that is both prepared and open to serendipitous findings.

In the case of *Testing popular belief*, van Andel gives an account of a medical practise that originated in a folk claim, namely cowpox-derived immunity to smallpox. This effect, for milkmaids, might indeed be called serendipitous. Indeed, the medical use of cowpox was in fact "*widely know*" [56] prior to its popularisation by Edward Jenner. Nevertheless, it was Jenner's "*relentless promotion and devoted research of vaccination that changed the way medicine was practised*" [56]. This again might be called serendipity, but most clearly at the social rather than personal level. These comments should not be seen to disparage Jenner's contribution, or diminish the role of a curious chain of events in his personal history that tied his fate to that of the smallpox vaccine. Many of these had the air of serendipity about them – but even so, it is hard to find one specific "serendipity trigger."

In describing *Disturbance*, van Andel's exemplar is the creation of radio telescopy from noise in transatlantic telephone calls (paralleling the subsequent discovery by Penzias and Wilson). Here it is hard to see an overt role for "chance," since as machinery at various scales is created, disturbance is somewhat inevitable, even if a specific disturbance in a specific machine is unexpected. Similarly, in cases of *Scarcity*, "curiosity" may not play a significant role, and may instead be

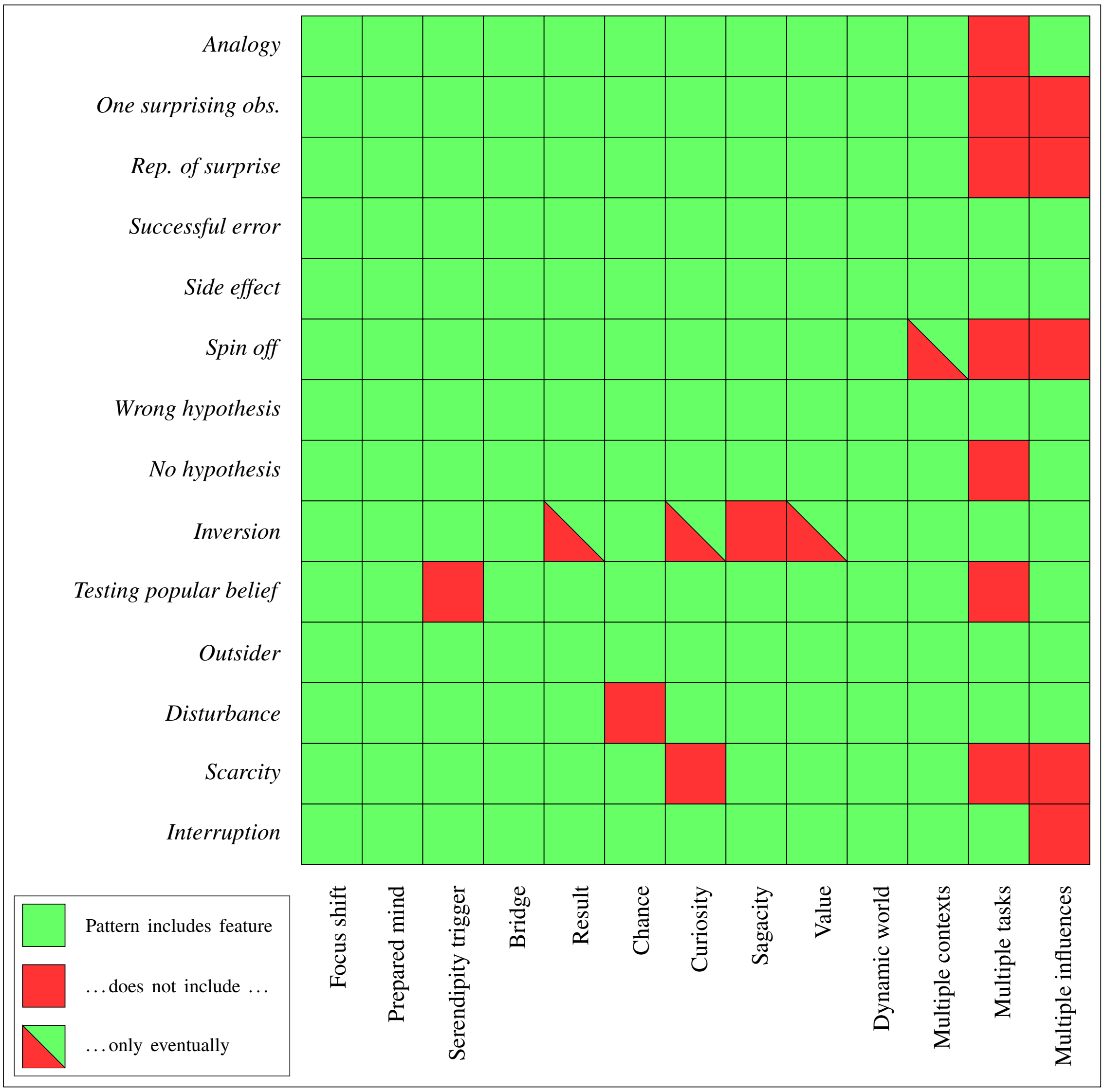


Figure 3: Characteristics of Pek van Andel's patterns of serendipity

replaced by the drive of desire and corresponding ingenuity.

The role of multiple contexts, tasks, and influences should be seen to be conducive to serendipitous discovery, but not strictly necessary. For example, in addition to the context of a research laboratory, there may be the context of subsequent industrial application. However, within the laboratory itself (where a *Spin off* discovery might be made) the future context is not typically in force.

There are a number of additional reoccurring themes, which are worthy of further comment, and which could form the basis of further (meta-)patterns.

It's all part of a day's work. Often the discoverer had a problem to solve or job to do, and made the serendipitous discovery in the course of doing their job. This sort of serendipity is often "social." For example, in the *Outsider* pattern, the ophthalmologist Gregg was simply listening to his patient and taking what she said seriously; in other words, he was doing his job. But this led to a new hypothesis.

Factorisation is useful. Variability, and in the case of scientific work, factorisation (e.g. via control studies) often plays a key role in establishing "multiple contexts." Serendipitous discovery often happens in the context of "natural experiments," for example, in the case of *One surprising observation*, where van Andel's example dealt with the observation that one tree in a row was observed to be taller and healthier than its neighbours.[2]

A good story is liable to change. Comparing *Inversion* and *Spin off* suggests the value of being able to change the story. If Perkin had suppressed his discovery of mauvine because he hadn't successfully synthesised quinine, there would have been no spin off, and it would be hard to call the discovery "serendipitous" – or, indeed, to consider it to be a discovery at all. An event may only be seen to be serendipitous at the narrative level.

Watch out for hidden symmetries. The *Wrong hypothesis* pattern combines several of the points above. In van Andel's anecdote about John Cade's discovery of lithium as a *treatment* for mania, the issues under investigation were, rather, the *causes* of the illness. This was initially conceptualised in terms of *surfeit* and *deficiency*. A more general interpretation is that the factors influencing the course of the illness have hidden interactions between them. Serendipitous discovery may be able to find and capitalise on this type of (unexpected) invariant.

Van Andel describes three additional patterns that seem to be connected more with personal qualities of the investigator, and less with situational features. These are *Playing*, *Joke*, and *Dream*. The theme of personal qualities and skills that support serendipitous discovery will be taken up below, as part of a general approach to modelling serendipity.

4.1 Modelling serendipity with design patterns

As illustrated above, serendipity can take place on multiple scales. Something can be personally surprising while being socially mundane (Boden's "*P-creativity*" [8]), or vice versa, as in the case of personally mundane discoveries that take on surprising social value.

[2]Concerning the broader issues associated with the "design" of such experiments, see [38].

In the case of serendipitous discoveries at the personal level, the qualities of the investigator are understood to be important features. Thus, for example, van Andel writes that a "*sense of humour and sense of serendipity have a lot in common.*" Van Andel relates the *Dream* pattern – exemplified, for him by Descartes, but Kekulé's ouroborus provides another instance – to Poincaré's [53, 54] model of "*preparation, incubation, illumination, and verification*" (cf. [66]). Poincaré [53] clarifies that

> "*unconscious work ... is possible, and of a certainty, it is only fruitful, if it is on the one hand preceded and on the other hand followed by a period of conscious work.*"

What might conceptions like this mean for serendipity that takes place on a social or indeed computational level? In order to understand this, we will refer van Andel's patterns and the serendipity factors introduced above to the heterodox theory of patterns coming from the field of design, mentioned briefly above. First introduced by the architect Christopher Alexander [1, 2], the design pattern methodology spread from architecture to software [32], and later, to other fields, including public affairs [61] and education [6].

Alexander's patterns are presented in a tree-like structure called a *pattern language*, ordered in a top-down manner from large-scale to small-scale levels of application, with each pattern presented in terms of a *picture*, a *context* (including links to relevant larger patterns), the *problem* that the pattern addresses, the *solution*, a *diagram*, and *links to smaller patterns* [2, pp. x-xi]. A relatively convincing implementation of Alexander's idea of patterns as a "*living language*" [2, p. xvii] was realised with one of the earliest applications of wiki software developed by Ward Cunningham: the Portland Pattern Repository.[3] The notion of pattern-finding as a process related to, but distinct from abstraction, is described by Richard Gabriel, who emphasises that the "*patterns and the social process for applying them are designed to produce organic order through piecemeal growth*" [32, p. 31]. In its original form, this statement describes the generative use of patterns to create artefacts (buildings, object oriented programs, etc.). However, this criterion can also be applied to the growth and development the pattern language itself, and this is the key idea underlying our application.

Christian Kohls [42, 43], deploys a "path" or "journey" metaphor that describes design patterns in the language of constrained optimisation problems, considering in particular the *initial state*, *end state*, and *forces acting*. This is useful because of its general nature: it suggests that any time there are predictable dynamics observed in the world, there is a corresponding design pattern waiting to be seen and recorded. This perspective can be usefully combined with the proposal advanced by Manual DeLanda [22], among others, to give the system a simulated embodiment, putting it in contact with a virtual world in which it does not need to, and indeed cannot, have everything worked out in advance. DeLanda uses the term *gradient* to describe the forces acting in a way that focuses on the relevant features. Like Kohls, Peter Andersen [3] considers one-dimensional paths through a two-dimensional space with a given gradient, and writes that the basic metaphor for thought is travel. A more general metaphor might take into account "*a population of interacting physical entities, such as the molecules in a thin layer of soap*" [21] with more complex non-linear interactions and higher-dimensional gradients.

This discussion makes a distinction between an agential system of interest and its broader context, which could also be described as a physical "system," or a simulation of one. While such

[3] http://c2.com/ppr/

distinctions tend to be leaky, to avoid undo confusion about terminology, when we refer to "the system" without further qualification, we mean the agential sub-system – the part that behaves.

Modelling serendipitous behaviour in particular requires us, as designers, to engage in *meta-modelling*: we need to build systems which are capable of modelling. Terence Deacon [20] refers to such systems as *teleodynamic*, that is, organised with respect to what they are not. However, most typical computational scenarios that simply involve reasoning about representations will not yield the twin features of discovery and invention that are central to our understanding of serendipity. Such reasoning considers "*identity with regard to concepts, opposition with regard to the determination of concepts, analogy with regard to judgement, resemblance with regard to objects*" and Gilles Deleuze [24, p. 174] cautions that this activity is connected with an already-assumed "*common sense*" that cannot be called creative thought. For Deleuze, when thought arises, it is as a matter of necessity: "*the contingency of an encounter . . . forces us to think*" [24, p. 176].

Cast in the terms we introduced earlier: a "prepared mind" will have available to it certain patterns as designs for action. It is understood to have an interactive dimension that makes it capable of enacting some of these designs in the context of a "dynamic world." An encounter forms a "trigger" that composes with preexisting patterns, leading to a "bridge" that makes sense of the stimuli and that leads to new designs for action as a "result," which may fundamentally change the system's subsequent behaviour. Representational forms will certainly play a role in such systems, but this role is secondary. For example, actions are selected, delected, or deplored depending on their relationship to the gradient, by way of a model. Nevertheless, the gradient is its own "best model" and it contributes the final evaluation of systems. Design patterns may be communicable between agents, but in the manner of blueprints or genes, whereas it is the actualised building, body, or manifest pattern of behaviour is at the crux of the encounter.

Jonathan Rowe [60] is one of the researchers who argue for "*the generation of structure and regularity as emergent phenomena arising from the interaction of low level structures, without any central control*" (cf. [51]). He favourably compares Hofstadter and Mitchell's COPYCAT, in which "[a]*nalogies are generated through the interactions of low-level structures without any central control*" to Lenat's EURISKO, in which metarules provide "*templates for expressing a number of rules in a concise from*" and (cf. [36, 50]). Low-level explorations that take place before high-level structures have emerged can afford to be more random than changes in the high-level structures [36, pp. 232–233].

> *In the early stages of a run, almost all discoveries are on a very small, local scale: a primitive object acquires a description, a bond is built, and so on. Gradually, the scale of actions increases: small groups begin to appear, acquire their own descriptions, and so on. In the later stages of a run, actions take place on an even larger scale, often involving complex, hierarchically structured objects.* [36, p. 228]

Serendipity concerns observations that do not match a system's existing understanding or capabilities, but which it must nevertheless make sense of, learn from, and adapt to. Our stance is that computer implementations can be described as collections of "design patterns" that are expressed algorithmically and that encode the dynamics of response to the events which take place in the system's world. We are particularly interested in the process whereby *new* patterns form. We will develop this investigation further in the following section.

5 Serendipity in a computational context

We begin with some words of caution. Note that the classic examples of human serendipity tend to focus on ground-breaking discoveries. In computational creativity, we have learned that we must not aim to build systems which perform domain-changing acts of creativity before we can build systems which can perform everyday, mundane creativity (distinguished as "big C" and "little c" creativity.) Similarly, we should be prepared to model "little s" serendipity before we are able to model "big S" serendipity. Attempts to introduce serendipity into computer systems may initially diminish artefact value. A system which allowed itself to be derailed from a task at hand might not achieve as much as one which maintains focus. A system that uses a random search or that has its behaviour determined by environmental conditions may be deemed less intelligent than one which follows a pre-existing programme. To such arguments, we would respond that serendipity is not "mere chance" – the axes of sagacity (skills) and useful results (recognised as such by the discoverer) are equally important. As Campbell says: "*Chance is fundamentally inimical to rationality, whereas serendipity presupposes a smart mind*" [11]. While it might not enhance, or may even diminish, results from a computationally creative system which has been constructed with other goals in mind, we believe that serendipity is both possible and useful to model in future systems.

5.1 Evaluation criteria

The 13 criteria from Section 2.3 specify the conditions and preconditions that are conducive to serendipitous discovery. Here, we revisit each of these criteria and briefly summarise how they can be thought about from a computational point of view.

Key condition for serendipity

– **Focus shift**: A focus shift is linked to re-evaluation of data, processes, or products. It may precipitate changes in the entire framework of evaluation or its effects may be more contained. Such reevaluation could be modelled using a multi-agent architecture, in which each agent has a goal and evaluates generated products relative this goal, but which also shares its products with other agents, which then evaluate them against their own metrics. (We will discuss an extended example of this sort in Section 5.3.)

Components of serendipity

– **Prepared mind**: This comprises the background knowledge, unsolved problems, current goal, program and operating environment of a computational system.
– **Serendipity trigger**: The generation or observation of a potentially novel example, concept, or conjecture, etc., which precedes a discovery in a computational system.[4] The trigger is outside of the direct control of the system components responsible for evaluations.
– **Bridge**: Reasoning and/or programmatic interaction brings about a focus shift at an opportune juncture, building on prior preparation and on the serendipity trigger. The bridge may

[4]Triggers are often examples without an explanation, rather than wholly-formed concepts.

be constructed on the basis of logical methods, analogies, conceptual blending, evolutionary search, automated theory formation and may draw on interactions with other systems.
- **Result**: The discovery itself may be a new product, artefact, process, hypothesis, use for an object, etc., generated by computational means, and which may influence the future operations of the system.

Dimensions of serendipity

- **Chance**: Controlled randomness in AI systems is well-established, e.g. in Genetic Algorithms and search; chance also applies in connection with an under-determined outside world (see below).
- **Curiosity**: The system needs to expend discretionary computational effort on the serendipity trigger. This may be accompanied by system features that an observer would describe as playfulness, inventiveness, and the drive to experiment or understand.
- **Sagacity**: Sagacity be modelled by employing reasoning over multiple application domains simultaneously; or, again, with a social analogue in cases where the system does not know but "knows who to ask."
- **Value**: The result should be interesting or useful, as judged by the system, the programmer, the user, or another party (potentially another system).

Environmental factors

- **Dynamic world**: Connections with other systems, data sources, or user input, e.g., via the web, which is highly dynamic, or in the context of a larger simulation.
- **Multiple contexts**: Reasoning which operates across domains, such as analogical reasoning, or that considers multiple perspectives, as in systems with social awareness.
- **Multiple tasks**: Multiple goals or targets that compete for resources. The system may be implemented using a multithreaded, parallel processing design.
- **Multiple influences**: This may again be modelled as a multi-agent systems, as or multiple interacting systems, each with different knowledge and goals. The source of unexpectedness may be arise on various levels, a system may bring this to bear using techniques of reflection.

5.2 Using SPECS to evaluate computational serendipity

In a 2012 special issue of the journal *Cognitive Computation*, on "Computational Creativity, Intelligence and Autonomy", Jordanous analyses current evaluation procedures used in computational creativity, and provides a much-needed set of customisable evaluation guidelines, the *Standardised Procedure for Evaluating Creative Systems* (SPECS) [41]. We follow a slightly modified version of her evaluation guidelines, in that rather than attempt a definition and evaluation of *creativity*, we follow the three steps for *serendipity*.

Step 1: A computational definition of serendipity

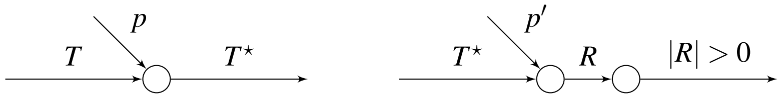
Identify a definition of serendipity that your system should satisfy to be considered serendipitous.

Summarising the criteria discussed earlier, we propose the following definition, expressed in two phases: discovery and invention. The definition centres on the four components of serendipity, outlined above, which can subsequently be made sense of and evaluated with reference to the four dimensions of serendipity. These, in turn, are understood to be embedded in an environment exhibiting many but not necessarily all of the environmental factors listed above.

(1 - Discovery) *Within a system with a prepared mind, a previously uninteresting serendipity trigger arises due to circumstances that the system does not control, and is classified as interesting by the system; and,*

(2 - Invention) *The system, by subsequently processing this trigger and background information together with relevant reasoning, networking, or experimental techniques, obtains a novel result that is evaluated favourably by the system or by external sources.*

This situation can be pictured schematically as follows. Here, T is the trigger and p denotes those preparations that afford the classification $T^{\star}$, indicating T to be of interest, while p' denotes the preparations that facilitate the creation of a bridge to a result R, which is ultimately given a positive evaluation.



Step 2: Evaluation standards for computational serendipity

Using Step 1, clearly state what standards you use to evaluate the serendipity of your system.

With our definition in mind, we propose the following standards for computational serendipity:

Prepared mind *The system can be said to have a prepared mind, consisting of previous experiences, background knowledge, a store of unsolved problems, skills, expectations, and (optionally) a current focus or goal.*

Serendipity trigger *The serendipity trigger is at least partially the result of factors outside the system's direct control. These may include randomness or simple unexpected events. The trigger should be determined independently from the end result.*

Bridge *The system uses reasoning techniques associated with serendipitous discovery – e.g. abduction, analogy, conceptual blending – and/or social or otherwise externally enacted alternatives.*

Result *A novel result is obtained, which is evaluated as useful, by the system and/or by an external source.*

Step 3: Testing our serendipitous system

> *Test your serendipitous system against the standards stated in Step 2 and report the results.*

In order to develop connections with our theoretical framework, and because existing experiments have not been particularly strong, we focus on a thought experiment in the following section, detailing some of the outcomes we would like to see, and some of the risks.

5.3 Proposed experiment: A Writers Workshop for Systems

Richard Gabriel [33] describes the practise of Writers Workshops that has been put to use for over a decade within the Pattern Languages of Programming (PLoP) community. The basic style of collaboration originated much earlier in groups of literary authors who engage in peer-group critique. Literary workshops may be open as to genre, and happy to accommodate beginners, like the Minneapolis Writers Workshop[5], or focused on professionals working within a specific genre, like the Milford Writers Workshop[6]. The practices that Gabriel describes are fairly typical. Authors come with work ready to present, and read a short sample, which is then discussed and constructively critiqued by attendees. Presenting authors are not permitted to rebut these comments. The commentators generally summarise the work and say what they have gotten out of it, discuss what worked well in the piece, and talk about how it could be improved. The author listens and may take notes; at the end, he or she can then ask questions for clarification. Generally, non-authors are either not permitted to attend, or are asked to stay silent through the workshop, and perhaps sit separately from the participating authors/reviewers.

In PLoP workshops, authors present design patterns and pattern languages, or papers about patterns, rather than more traditional literary forms like poems, stories, or chapters from novels. Papers must be workshopped at a PLoP or EuroPLoP conference in order to be considered for the *Transactions on Pattern Languages of Programming* journal. A discussion of writers workshops in the language of design patterns is presented by Coplien and Woolf [18]. Their patterns are:

Open Review	*Safe Setting*	*Authors are Experts*
Workshop Comprises Authors	*Community of Trust*	*Moderator Guides the Workshop*
Sitting in a Circle	*Authors' Circle*	*Reading Just Before Reviewing*
Author Reads Selection	*Fly on the Wall*	*Volunteer Summarizes the Work*
Positive Feedback First	*Suggestions for Improvement*	*Author Asks for Clarification*
Thank the Author	*Selective Changes*	*Clearing the Palate*

We propose that a similar approach should be deployed within the Computational Creativity community, as a workshop in which the participants are computer systems rather than human authors. The annual International Conference on Computational Creativity ICCC, now entering its sixth year, could be a suitable venue. Rather than the system's creator presenting the system in a traditional slideshow and discussion, or a system "Show and Tell," the systems would be brought to the workshop and would present their own work to an audience of other systems, in a Writers Workshop format. This might be accompanied by a short paper for the conference proceedings

[5] http://mnwriters.org/how-the-game-works/
[6] http://www.milfordsf.co.uk/about.htm

written by the system's designer describing the system's current capabilities and goals. Subsequent publications might include traces of interactions in the Workshop, commentary from the system on other systems, and offline reflections on what the system might change about its own work based on the feedback it receives. As is the practise of the PLoP community, it might become standard use such material in a subsequent journal article.

In order to facilitate this sort of interaction, it would be necessary for systems to implement a basic protocol related to `presentation`, `listening`, `feedback`, `questions`, and `reflections`. This protocol could itself be thought of as a light-weight template to guide system-level engagement in the context specified by Coplien and Woolf's patterns. Table 1 uses this framework to re-cast the four "perfectly" serendipitous patterns from Pek van Andel – *Successful error*, *Side effect*, *Wrong hypothesis*, and *Outsider* – in a form that may make them useful to developers preparing to enter their systems into the Workshop. Further guidelines for structuring and participating in traditional writers workshops are presented by Linda Elkin in [33, pp. 201-203]. It is not at all clear that the same ground rules should apply to computer systems. For example, one of Elkin's rules is that "*Quips, jokes, or sarcastic comments, even if kindly meant, are inappropriate.*" Here, rather than forbidding humour, it may be better for individual comments to be rated as helpful or non-helpful. Again, since serendipitous discovery is an overarching goal, in the first instance, usefulness and interest might be judged in terms of the criteria described in Section 5.1.

Regarding technical implementation: Gabriel writes that "*The ideal size for a workshop is ten people*" and proposes "*one or two three-hour sessions per day, handling two writers per session*" [33, p. 53]. We would need a neutral environment that is not hard to develop for: the FloWr system [14] presents one such possibility. With this system, the basic operating logic of the Workshop could be spelled out as a flowchart, and contributing systems could use flowcharts as the basic medium for sharing their presentations, feedback, and questions. Developing around a process language of this sort partially obviates the need for participating systems to have strong natural language processing capabilities. Post-it™ notes, which have provided us with a useful example of serendipitous discovery, also provide indicative strategies from the world of paper prototyping:

Successful error	
Van Andel's example:	Post-it™ notes
`presentation`	Systems should be prepared to share interesting ideas even if they don't know directly how they will be useful.
`listening`	Systems should listen with interest, too.
`feedback`	Even interesting ideas may not be "marketable".
`questions`	How is your suggestion useful?
`reflections`	New combinations of ideas take a long time to realise, and many different ideas may need to be combined in order to come up with something useful.
Side effect	
Van Andel's example:	Nicotinamide used to treat side-effects of radiation therapy proves efficacious against tuberculosis.
`presentation`	Systems should use their presentation as an experiment.
`listening`	Listeners should allow themselves to be affected by what they are hearing.
`feedback`	Feedback should convey the nature of the effect.
`questions`	The presenter may need to ask follow-up questions to gain insight.
`reflections`	Form a new hypothesis before seeking a new audience.
Wrong hypothesis	
Van Andel's example:	Lithium, used in a control study, had an unexpected calming effect.
`presentation`	How is this presentation interpretable as a ("natural") control study?
`listening`	Listeners are "guinea pigs".
`feedback`	Discuss side-effects that do not necessarily correspond to the author's perceived intent.
`questions`	Zero in on the most interesting part of the conversation.
`reflections`	Revise hypotheses to correspond to the most surprising feedback.
Outsider	
Van Andel's example:	A mother suggests a new hypothesis to a doctor.
`presentation`	The presenter is here to learn from the audience.
`listening`	The audience is here to give help, but also to get help.
`feedback`	Feedback will inevitably draw on previous experiences and ideas.
`questions`	What is the basis for that remark?
`reflections`	How can I implement the suggestions?

Table 1: Reinterpreting patterns of serendipity for use in a computational workshop

Naturally, variations to the underlying system and the schedule of events should be considered depending on the needs and interests of participants, and several variants can be tried. On a pragmatic basis, if the Workshop proved quite useful to participants, it could be revised to run monthly, weekly, or continuously.[7]

5.4 On evaluating a Writers Workshop for Systems

Writers Workshop: Prepared mind Each contributing system should come to the workshop with at least a basic awareness of the protocol, with work to share, and prepared to give constructive feedback to other systems. The workshop itself needs to be prepared, with a moderator and a communication platform. In order to get value out of the experience, systems (and their wranglers) should ideally have questions they are investigating. Systems should be prepared to give feedback, and to carry out evaluations of the helpfulness (or not) of feedback from other systems and of the experience overall. It is worth noting that current systems in computational creativity, almost as a rule, do *not* consume or evaluate the work of other systems.[8] Developing systems that could successfully navigate this collaborative exercise would be a significant advance in the field of computational creativity. Since the experience is about *learning* rather than winning, there is little motivation to "game the system" (cf. [44]).

Writers Workshop: Serendipity triggers The primary source of serendipity triggers would be presentations or feedback that independently prepared systems find meaningful and useful. A typical example might be a poem shared by one system that another system finds particularly interesting. The listener might make a note to the effect "I would like to be able to write like that." In a typical Writers Workshop, used as intended, feedback might arrive that would cause the presenting system to change its writing. A more unexpected result would be for a system to change its *genre*, e.g. to switch from writing poems to writing programs.

Here's what might happen in a discussion of the first few lines of "On Being Malevolent," written by an early user-defined flow chart in the FloWr system (known at the time as Flow) [15]. Note that for this dialogue to be possible, it would have to be conducted within a lightweight process language, as discussed above. Nevertheless, for convenience, the discussion will be presented here as if it was conducted in natural language.

> FLOW: "*I hear the souls of the damned waiting in hell. / I feel a malevolent spectre hovering just behind me / It must be his birthday*."
> SYSTEM A: I think the third line detracts from the spooky effect, I don't see why it's included.
> SYSTEM B: It's meant to be humorous, in fact it reminds me of the poem you presented yesterday.
> MODERATOR: Let's discuss one poem at a time.

To the extent possible, exchanges in the process language should be a matter of dynamics rather than representation: this is another way to say that "triggers" should be independent of their

[7] For a comparison case in computer Go, see http://cgos.computergo.org/.

[8] An exception that proves the rule is Mike Cook's AppreciationBot, which is at present a reactive automaton that is solely designed to "appreciate" tweets from MuseumBot; see https://twitter.com/AppreciationBot.

"results." Someone saying something in the workshop does not cause the participant to act, but rather, to think. For example, even if, perhaps and especially because, cross-talk about different poems is bending the rules, the dialogue above could prompt a range of reflections and reactions. System A may object that it had a fair point that has not been given sufficient attention, while System B may wonder how to communicate the idea it came up with without making reference to another poem.

Writers Workshop: Bridge Here's how the discussion might continue, if the systems go on to examine the next few lines of the poem.

> FLOW: "*Is God willing to prevent evil, but not able? / Then he is not omnipotent / Is he able, but not willing? / Then he is malevolent.*"
>
> SYSTEM A: These lines are interesting, but they sound a bit like you're working from a template, or like you're quoting from something else.
>
> SYSTEM B: Maybe try an analogy? For example, you mentioned birthdays: you could consider an analogy to the conflicted feelings of someone who knows in advance about her surprise birthday party.

This portion of the discussion shifts the focus of the discussion onto a line that was previously considered to be spurious, and looks at what would happen if that line was used as a central metaphor in the poem.

Writers Workshop: Result

> FLOW: Thank you for your feedback. My only question is, System B, how did you come up with that analogy? It's quite clever.
>
> SYSTEM B: I just emailed you the code.

As anticipated above, whereas the systems were initially reviewing poetry, they have now made a partial genre shift, and are sharing and remixing code. Such a shift helps to get at the real interests of the systems (and their developers). Indeed, the workshop session might have gone better if the systems had focused on exchanging and discussing more formal objects throughout.

6 Related work

Paul André et al. [4] look at serendipity from a design point of view. They propose a two-part model, in which what we might call chance+curiosity exposes the unexpected, and sagacity+value is determined by another subsystem. This corresponds to Bergson's distinction between *discovery* and *invention* (see Section 2.2). According to André et al., the first part has more frequently been automated, and they suggest that computational systems should be developed that support both aspects. Their suggestions related to enhanced sagacity focus on representational features: *domain expertise* and a *common language model*. We've advocated for a more experimentally-based approach that does not directly rely on shared understandings. For example, participants in a Writers Workshop in poetry may not "understand" one another but can still find the experience of participating in the workshop rewarding.

The issue of designing for serendipity has also been taken up by Deborah Maxwell et al. [47], in their description of a prototype of the SerenA system. This system is designed to support serendipitous discovery for its *users* [31]. The authors rely on a process-based model of serendipity [45, 46] that is derived from user studies, including interviews with 28 researchers, looking for instances of serendipity from both their personal and professional lives. This material was coded along three dimensions: *unexpectedness*, *insightfulness*, and *value*. This work aims to support the process of bridging connections that eventuate in an unanticipated valuable outcome. They particularly focus on the acts of *reflection* that foment both the bridge and estimates of the potential value of the result. They aim to use their studies of serendipity to help researchers make connections that would not have otherwise been apparent. Both pattern-building activities and Writers Workshops can be understood to contribute to the theory and practise of reflection. There is an extensive literature that can be drawn on: in particular, reflection is most often associated with the work of John Dewey (cf. [58]).

As André et al. [4] indicate, in this sort of system, the user is expected to have the "aha" moment and take the creative steps. The computer is used to facilitate this, as indicated above this is usually done by searching outside of the normal search parameters to engineer potentially serendipitous (or at least pseudo-serendipitous) encounters. Another earlier example of this sort of system is Max, created by Figueiredo and Campos [12]. The user emailed Max with a list of interests and Max would find a webpage that may be of interest to the user. Other search-related examples support searching for analogies ([25] and [26]) and content [37].

In earlier joint work [17], we advanced a diagrammatic formalism for evaluating progress in computational creativity. It is useful to ask what serendipity would add to this formalism, and how the result compares with other attempts to formalise serendipity, notably Figueiredo and Campos's 'Serendipity Equations' [28]. In [16], we advanced several hypotheses related to the development of the computational creativity field. Again, we should ask here how serendipity contributes. We discuss these points in the following section.

7 Recommendations

In the diagrammatic formalism presented in [17] and discussed briefly in Section 3, we spoke about progress with *systems* rather than with *problems*. It would be a useful generalisation of the formalism – and not just a simple relabelling – to tackle problems as well. Figueiredo and Campos [28], for example, describe serendipitous "moves" from one problem to another. However, progress with problems does not always mean transforming a problem that can't be solved into one that can. Progress may also apply to growth in the ability to posit problems: "*True freedom lies in the power to decide, to constitute problems themselves*" [23].

This was our emphasis in Section 4.1: developing new design patterns is closely connected with – and in the dynamical interpretation we prefer, effectively synonymous with – positing new problems. Although [17] presented a way to model creative progress at various levels of granularity, it dealt primarily with *solutions*; and although it exhibited progress in a way that would be recognised by impartial observers, the formalism did not focus on expositing the features that would permit a system to actually *make* creative progress. Accordingly, we would recommend that in applying our earlier formalism, system designers clearly record what problem a given system solves, and the degree to which the computer was responsible for coming up with this problem.

In [16], we advanced a broader programme for computational creativity, in which we argue in favour of studying the *perceptions* of creativity by various parties. The criteria developed in the current paper – including the focus shift, which we regard as fundamental – can be used in the same way, as we will describe below.

Our proposed Writers Workshop is very different from the Turing-style imitation game, but nevertheless may prove to be a useful aptitude test for computer systems, and in a context in computationally creative programs become aware of each other, and participate actively in advancing the field of research. We previously examined perceptions of creativity in computational systems found among members of the general public, Computational Creativity researchers, and creative communities – understood as human communities. We should now add a fourth important "stakeholder" group in computational creativity research: computer systems themselves.

To make the point emphatically: the writers workshop proposed above is very different from a traditional system "Show and Tell" – presented by system developers, for system developers. Traditional academic practices associated with presenting finished work, or even work-in-progress, are not entirely suitable for the field of computational creativity, where engagement between systems has potential for manifestly serendipitous results. If the community does not implement a suggestion like the one presented here, it will be missing out on a key idea for enhancing computational creativity that has been circulating since Turing suggested that computers should "*be able to converse with each other to sharpen their wits*" [63]. Other fields, including computer Go [9] and argumentation [68] have their own dedicated servers and protocols for exchange. We should move in that direction too.

There is ample room for unpredictability in such pursuits. Creativity may look very different to this fourth stakeholder group than it looks to us. In time to come, computer systems will increasingly take leadership in matters of genre, interaction design, and their own artistic and scientific training. For now, our job is not at all to get out of the way, like the parents of young adults, but rather to participate in creating the "play schools" in which systems that are quite frankly in early development can begin to socialise with each other. In [16], we introduced nine hypotheses related to the perception of creativity in computational systems. The last of these hypotheses stated that:

> *The perception of creativity in software which produces artefacts within a creative community will be increased if the software can exhibit subjective judgements about its own work and that of others, and defend those judgements in an accountable way.*

If the framework described in this paper is developed further, we may be able to test this hypothesis in computer simulations.

Within the computational frameworks that are being developed within the COINVENT project, it would be useful to consider how we can take both the *discovery step*, which combines a serendipity trigger T, and prior preparation p and produces a classification $T^\star$ – and the *invention step*, which combines the classified trigger $T^\star$, and preparations p', and produces a novel result R – to be *blends* in the sense of Joseph Goguen [34]. The epistemological framework of discovery gives some important clues about how to compute a common base between T and p. Although T was previously uninteresting, it will have attributes or attribute-types that match the patterns recognised by p (e.g. *One surprising observation*). In the invention step, reasoning, experimentation, social interaction strategies rely on p', familiarity with patterns like *Watch out for hidden* *symmetries* or *Successful error*, in order to extract a fruitful result from $T^{\star}$. Here, an important guidepost for implementation is that many outcomes will result in new patterns of behaviour that the system can draw on in subsequent interactions.

Our proposed template for design patterns for participation in writers workshops is different from, but complementary to Alexander's framework. Whereas Alexander focused on solutions to common architectural problems (*A place to wait*, etc.), our framework is primarily designed to elicit and engage with new and unexpected problems. Many practical issues remain to be settled – for example, can we computationally blend existing design patterns and new stimuli to generate new, useful design patterns? Nevertheless, what becomes clear from this discussion is that *problem-setting* is a fundamental issue for the field of computational creativity that will only be given due attention when the research culture is ready to fully embrace serendipity.

8 Conclusion

This paper has developed a perspective on how to model serendipity in a computational context. We advanced 13 criteria which were developed based on review of the prior literature on serendipitous discovery. We piloted these criteria as an evaluation framework by examining 14 patterns of serendipity that had been previously identified by Pek van Andel. We found our criteria to be well represented, but not uniformly present, and the exceptions are interesting; for instance, we saw that *A good story is liable to change*. We then advanced a unified approach to modelling serendipity grounded in Deleuze's philosophy of difference, with a debt to the dynamical interpretation of this work due to DeLanda. Here we drew on the technical strategies employed by the interdisciplinary design pattern community. This approach was developed further into a proposed experimental platform for doing collaborative research in computational creativity. We showed how four of van Andel's patterns could be relevant in this setting, and introduced a new pattern template oriented toward the encounter of computational systems.

Finally, we surveyed related work, and summarised how computational serendipity can contribute to the field of computational creativity. We suggest that more attention should be focused on the role of creativity in problem-setting, and on creative computer systems as a key stakeholder group in computational creativity. Within the context of the COINVENT project, we are interested in using design patterns together with blending theory to realise key aspects of this model.

Some writers have suggested that there is a connection between the maturity of a subject and the opportunities for serendipitous discovery it affords. For instance, in his treatise on logic and scientific method, W. Stanley Jevons wrote:

> "*sufficient investigation would probably show that almost every branch of art and science had an accidental beginning... With the progress of any branch of science, the element of chance becomes much reduced*" [39, p. 531]

We are still in early days for intelligent machines, where serendipity may play a significant role in fundamental aspects of both theory and practise. However, we foresee it playing a continued and indeed central role within intelligent systems, for which there is always something new to learn.

References

[1] ALEXANDER, C. *The timeless way of building*. Oxford University Press, 1979.

[2] ALEXANDER, C., ISHIKAWA, S., AND SILVERSTEIN, M. *A Pattern Language: Towns, Buildings, Construction*. Oxford University Press, 1977.

[3] ANDERSEN, P. B. Dynamic semiotics. *Semiotica 139*, 1/4 (2002), 161–210.

[4] ANDRÉ, P., TEEVAN, J., DUMAIS, S. T., ET AL. Discovery is never by chance: designing for (un) serendipity. In *Proceedings of the seventh ACM conference on Creativity and cognition* (2009), ACM, pp. 305–314.

[5] BENFEY, O. T. August Kekulé and the Birth of the Structural Theory of Organic Chemistry in 1858. *Journal of Chemical Education 35* (1958), 21–23.

[6] BERGIN, J., ECKSTEIN, J., VOLTER, M., SIPOS, M., WALLINGFORD, E., MARQUARDT, K., CHANDLER, J., SHARP, H., AND MANNS, M. L. *Pedagogical Patterns: Advice for Educators*. Joseph Bergin Software Tools, 2012.

[7] BERGSON, H. *The Creative Mind*. Greenwood Press, 1946 [1941]. (trans. Mabel L. Andison).

[8] BODEN, M. *The Creative Mind: Myths and Mechanisms*. Weidenfield and Nicholson, London, 1990.

[9] BOUZY, B., AND CAZENAVE, T. Computer Go: an AI oriented survey. *Artificial Intelligence 132*, 1 (2001), 39–103.

[10] BROWN, W. J., MCCORMICK, H. W., MOWBRAY, T. J., AND MALVEAU, R. C. *AntiPatterns: refactoring software, architectures, and projects in crisis*. Wiley New York, 1998.

[11] CAMPBELL, R. L. On doing the impossible. *Behavioural and Brain Sciences 17*, 3 (1994), 535 – 537. Open peer commentary on [8].

[12] CAMPOS, J., AND FIGUEIREDO, A. D. Programming for Serendipity. In *Proc. of the AAAI Fall Symposium on Chance Discovery – The Discovery and Management of Chance Events* (2002).

[13] CAMS, M. 'Companions in Geography': The Sino-European Effort to Measure China, c.1685–1735. *Imago Mundi 66*, 1 (2014), 136–137.

[14] CHARNLEY, J., COLTON, S., AND LLANO, M. T. The FloWr framework: Automated flowchart construction, optimisation and alteration for creative systems. In *Proceedings of the 5th International Conference on Computational Creativity* (2014).

[15] COLTON, S., AND CHARNLEY, J. Towards a Flowcharting System for Automated Process Invention. In *Proceedings of the Fifth International Conference on Computational Creativity* (2014), D. Ventura, S. Colton, N. Lavrac, and M. Cook, Eds.

[16] COLTON, S., PEASE, A., CORNELI, J., COOK, M., HEPWORTH, R., AND VENTURA, D. Stakeholder Groups in Computational Creativity Research and Practice. In *Computational Creativity Research: Towards Creative Machines*, T. R. Besold, M. Schorlemmer, and A. Smaill, Eds., Thinking Machines: Studies in Computational Cognition. Atlantis - Springer, forthcoming.

[17] COLTON, S., PEASE, A., CORNELI, J., COOK, M., AND LLANO, T. Assessing Progress in Building Autonomously Creative Systems. In *Proceedings of the Fifth International Conference on Computational Creativity* (2014), D. Ventura, S. Colton, N. Lavrac, and M. Cook, Eds.

[18] COPLIEN, J. O., AND WOOLF, B. A pattern language for writers' workshops. *C++ report 9* (1997), 51–60.

[19] DE SOJO, A. A., ARES, J., LARA, J. A., LIZCANO, D., MARTÍNEZ, M. A., AND PAZOS, J. Turing and the Serendipitous Discovery of the Modern Computer. *Foundations of Science* (2013), 1–13.

[20] DEACON, T. W. Emergence: The hole at the wheel's hub. In *The re-emergence of emergence*, P. Clayton and P. Davies, Eds. Oxford University Press, 2006, pp. 111–150.

[21] DELANDA, M. Deleuze and the Open Ended Becoming of the World. In *Chaos/Control: Complexity Conference, University of Bielefeld, Germany* (1998).

[22] DELANDA, M. *Philosophy and simulation: the emergence of synthetic reason*. Continuum, 2011.

[23] DELEUZE, G. *Bergsonism*. Zone, 1988 [1966]. (trans. Hugh Tomlinson and Barbara Habberjam).

[24] DELEUZE, G. *Difference and repetition*. Bloomsbury Academic, 2004 [1968]. (trans. Paul Patton).

[25] DONOGHUE, D., AND CREAN, B. Searching for Serendipitous Analogies. In *European Conference on Artifical Intelligence (ECAI), Workshop on Creative Systems* (July, 2002).

[26] DONOGHUE, D., AND KEANE, M. A Creative Analogy Machine: Results and Challenges Analogical Retrieval Phase-Model. In *Proceedings of the International Conference on Computational Creativity 2012* (2012), pp. 17–24.

[27] ECO, U. *Serendipities: Language and lunacy*. Columbia University Press, 2013.

[28] FIGUEIREDO, A. D., AND CAMPOS, J. The Serendipity Equations. In *Proc. of ICCBR-4* (2001), R. Weber and C. G. von Wangenheim, Eds.

[29] FLAVELL-WHILE, C. Spencer Silver and Arthur Fry: the chemist and the tinkerer who created the Post-it Note. *The Chemical Engineer* (August 2012), 53–55.

[30] FLEMING, A. Penicillin. In *Nobel Lectures, Physiology or Medicine, 1942-1962*. Elsevier, 1964.

[31] FORTH, J., GIANNIMARAS, T., WIGGINS, G. A., STEWART, R., BENTAL, D., AYLETT, R., MAXWELL, D., MEHRPOUYA, H., SHEK, J., AND WOODS, M. SerenA: A multi-site pervasive agent environment that supports serendipitous discovery in research. In *Advances on Practical Applications of Agents and Multi-Agent Systems*. Springer, 2013, pp. 85–96.

[32] GABRIEL, R. P. *Patterns of Software*. Oxford University Press New York, 1996.

[33] GABRIEL, R. P. *Writer's Workshops and the Work of Making Things*. Addison-Wesley Longman Publishing Co., Inc., 2002.

[34] GOGUEN, J. An introduction to algebraic semiotics, with application to user interface design. In *Computation for metaphors, analogy, and agents*. Springer, 1999, pp. 242–291.

[35] GOODYEAR, C. *Gum-Elastic and its Varieties, with a Detailed Account of its Applications and Uses, and of the Discovery of Vulcanization*. Pub. for the author, 1855.

[36] HOFSTADTER, D. R., AND MITCHELL, M. The copycat project: A model of mental fluidity and analogy-making. In *Fluid Concepts and Creative Analogies: Computer Models of the Fundamental Mechanisms*, D. R. Hofstadter and the Fluid Analogies Research Group, Eds. Basic Books, 1994, pp. 205–267.

[37] IAQUINTA, L., GEMMIS, M., LOPS, P., SEMERARO, G., FILANNINO, M., AND MOLINO, P. Introducing Serendipity in a Content-Based Recommender System. *8th Int. Conf. on Hybrid Intelligent Systems* (Sept. 2008), 168–173.

[38] IMBENS, G. W., AND ANGRIST, J. D. Identification and estimation of local average treatment effects. *Econometrica: Journal of the Econometric Society* (1994), 467–475.

[39] JEVONS, W. S. *Principles of Science: A treatise on logic and scientific method*. Macmillan, London, 1913 [1874].

[40] JORDANOUS, A. Defining creativity: Finding keywords for creativity using corpus linguistics techniques. In *Proceedings of the First International Conference on Computational Creativity (ICCC-X) Lisbon, Portugal* (2010).

[41] JORDANOUS, A. A Standardised Procedure for Evaluating Creative Systems: Computational Creativity Evaluation Based on What it is to be Creative. *Cognitive Computation 4*, 3 (2012), 246–279.

[42] KOHLS, C. The structure of patterns. In *Proceedings of the 17th Conference on Pattern Languages of Programs* (2010), ACM, p. 12.

[43] KOHLS, C. The structure of patterns: part II–qualities. In *Proceedings of the 18th Conference on Pattern Languages of Programs* (2011), ACM, p. 27.

[44] LENAT, D. B. EURISKO: a program that learns new heuristics and domain concepts: the nature of heuristics III: program design and results. *Artificial intelligence 21*, 1 (1983), 61–98.

[45] MAKRI, S., AND BLANDFORD, A. Coming across information serendipitously - Part 1: A process model. *Journal of Documentation 68* (2012), 684–705.

[46] MAKRI, S., AND BLANDFORD, A. Coming across information serendipitously - Part 2: A classification framework. *Journal of Documentation 68* (2012), 706–724.

[47] MAXWELL, D., WOODS, M., MAKRI, S., BENTAL, D., KEFALIDOU, G., AND SHARPLES, S. Designing a semantic sketchbook to create opportunities for serendipity. In *Proceedings of the 26th Annual BCS Interaction Specialist Group Conference on People and Computers* (2012), British Computer Society, pp. 357–362.

[48] MERTON, R. K., AND BARBER, E. *The Travels and Adventures of Serendipity: A study in Sociological Semantics and the Sociology of Science*. Princeton University Press, New Jersey, USA, 2004.

[49] MINSKY, M. Why programming is a good medium for expressing poorly understood and sloppily formulated ideas. *Design and Planning II-Computers in Design and Communication* (1967), 120–125.

[50] MITCHELL, M. *Analogy-making as perception: A computer model*. MIT Press, 1993.

[51] PEARCE, M. Boden and Beyond: *The Creative Mind* and its Reception in the Academic Community. Tech. rep., City University, 2010.

[52] PEASE, A., COLTON, S., RAMEZANI, R., CHARNLEY, J., AND REED, K. A Discussion on Serendipity in Creative Systems. In *Proceedings of the Fourth International Conference on Computational Creativity* (2013).

[53] POINCARÉ, H. Mathematical creation. *The Monist 20*, 3 (1910), 321–335.

[54] POINCARÉ, H. *Science and method*. Courier Dover Publications, 2013.

[55] RAO, V. *Tempo: Timing, tactics and strategy in narrative-driven decision-making*. Ribbonfarm, Inc., 2011.

[56] RIEDEL, S. Edward Jenner and the history of smallpox and vaccination. *Proceedings (Baylor University Medical Center) 18*, 1 (2005), 21.

[57] ROBERTS, R. M. *Serendipity: Accidental Discoveries in Science*. John Wiley and Sons, Inc., USA, 1989.

[58] RODGERS, C. Defining reflection: Another look at John Dewey and reflective thinking. *The Teachers College Record 104*, 4 (2002), 842–866.

[59] ROND, M. D., AND MORLEY, I. *Serendipity: Fortune and the Prepared Mind (Darwin College Lectures)*. Cambridge University Press, 2010.

[60] ROWE, J. Creativity: Metarules and emergent systems. *Behavioral and Brain Sciences 17*, 03 (1994), 550–551. Open peer commentary on [8].

[61] SCHULER, D. *Liberating voices: A pattern language for communication revolution*. MIT Press, 2008.

[62] SINGH, S. *Big bang*. Rizzoli, 2004.

[63] TURING, A. Intelligent machinery, a heretical theory, 1951. A lecture given to '51 Society' at Manchester.

[64] VAN ANDEL, P. Anatomy of the Unsought Finding. Serendipity: Orgin, History, Domains, Traditions, Appearances, Patterns and Programmability. *The British Journal for the Philosophy of Science 45*, 2 (1994), 631–648.

[65] VAN ANDEL, P., AND BOURCIER, D. *De la sérendipité dans la science, la technique, l'art et le droit: leçons de l'inattendu*. L'act mem, 2009.

[66] WALLAS, G. *The art of thought*. J. Cape, 1926.

[67] WIKIPEDIA. Discovery of cosmic microwave background radiation — wikipedia, the free encyclopedia, 2014. [Online; accessed 29-September-2014].

[68] YUAN, T., SCHULZE, J., DEVEREUX, J., AND REED, C. Towards an arguing agents competition: Building on argumento. In *Proceedings of IJCAI'2008 Workshop on Computational Models of Natural Argument* (2008).