Towards Viable Computer Systems: a Set Theory Interpretation of Ecological Dependence within Beer’s Self-Organizing Viable System Model

R.J. Thompson  A.G. Laws  D.J. Reilly  A. Taleb-Bendiab  D. Llewellyn-Jones
School of Computing & Mathematical Sciences
Liverpool John Moores University, United Kingdom L3 3AF
(+44) (0)151 231 2280
(+44) (0)151 231 2270
(+44) (0)151 231 2279
(+44) (0)151 231 2489
(+44) (0)151 231 2270
R.J.Thompson@ljmu.ac.uk
A.Laws@ljmu.ac.uk
D.Reilly@ljmu.ac.uk
A.TalebBendiab@ljmu.ac.uk
D.Llewellyn-Jones@ljmu.ac.uk

ABSTRACT
Presented is research articulating a novel technology progressing resource management within self-organizing systems. Examining both Cybernetic and Autonomic Computing techniques we evolve a set-theory oriented, atomically-derived, emergent model that reflects an algorithmic decomposition of Beer's recursive, multi-agent Viable System Model, pertinent by its composition of multiple and independent entities, sharing one or more objectives. Integrated management promotes each sub-system as a whole within a closed ecological meta-boundary. The relationships between sub-systems is demonstrated via syntax subscripts, while the relationship linking recursive levels is recognized via superscripts. The resultant design grammar endorses autonomy versus governance, exploiting cybernetic, biological and mathematical metaphors, crucially seeking inherent learning and control through system-environment interplay. Focusing on interactions and inter-relationships, the self-organizing environments exhibit evolution of systemic elements, conserving yet managing resources provided by each entity. Research ultimately aspires augmentation of the Autonomic Computing state of the art into the original field of Viable Computing Systems.

Categories and Subject Descriptors
I.2.11 [Artificial Intelligence]: Distributed Artificial Intelligence – intelligent agents, multiagent systems.

General Terms

Keywords
Self-organizing systems, ecological dependence, Autonomic Computing, Viable System Model, Cybernetics, Agent-Based Modelling.

1. INTRODUCTION
Increasing software system complexity [1, 2] is recognized as a great contemporary challenge of the software industry, with Autonomic Computing [3] being proposed as a resolution. This study seeks to advance the state of that art. Our research formerly combined a priori biological [4, 5] and cybernetic [6] metaphors to create a mathematical analogue [7] of the key functionality of Beer’s multi-agent Viable System Model (VSM) [8]. Our work has progressed to innovatively model the relationship between the VSM recursions, via an algebraic set-theory blueprint; the foundation of an emerging design grammar model. In this paper we emphasize the importance of system-environment inter-dependence toward this end. The eventual research goal is to manifest a self-organizing Viable Computer System (VCS) corresponding to the subsumption of human autonomic systems by cognitive systems. Emergence will preserve its’ viability, that is, its ability to exist in a changing environment, reducing redundancy and thus complexity. The VCS will respond to environmental stimulus post time t\textsuperscript{n-1}, intrinsically reacting, whilst forecasting. Our research will apply both homeostatic [9], and autopoietic [10, 11] approaches to generate a referential self model of the internal systemic capabilities \textsuperscript{t} and a model reflecting the required world situation \textsuperscript{t+1}, thereby seeking to autonomically address environmental requirements, via feedback control.

Having previously reformulated the relationships between the systems and their respective environments of the VSM [7], the current work progresses to articulate a context-free Design Grammar model depicting the architectural recursivity, or superscript relationships. This syntax mirrors the VSM holism within each implementation system, \textit{S\textsubscript{i}} and the associated environments. Both models recurse and emerge bottom-up, offering the potential for modelling a dynamic system providing an internal representation and an architecture imparting self-awareness [12] by reflecting environmental ecology. We advance the VCS, supporting extension of the recursivity within a previous cybernetic architecture [13].

The remainder of the paper is structured as follows: Section II considers the state of the art; cybernetics vis-à-vis Autonomic Computing. Section III summarizes Beer’s Viable System Model, and how our research has translated the context into that of the VCS. Section IV describes our Design Grammar Model system-environment ecology and the novel superscript identities syntax, whilst reviewing our unique subscript Design Grammar Model and its own respective ecology. Section V draws conclusions from the research to date.
2. AUTONOMIC COMPUTING VIS A VIS CYBERNETICS

The launch of the IBM Autonomic Computing initiative in 2001 [14] led Horn to voice the rising problems of complexity in software systems; chiefly in managing both the growth and maintainability of code, so averting Legacy System Syndrome [1]. Autonomic Computing systems self-regulate, modifying their actions in response to a changing environment, akin to the self-management of biological systems. Previously, Laws et al. in 1999 proposed applying the cybernetic qualities of Beer’s VSM to address this complexity [15]. Associating this to concerns later accepted as pertinent to the Autonomic Computing ideal, this research produced the 2001 J-Reference Model [13]. Merging Beer's topology and Brutman et al.'s IRMA [16] with the Beliefs, Desires, Intentions (BDI) theory, the J-Reference Model [1] applied Ashby’s variety [6] concept to the ideal. The work concluded, that applying a real-time, context-sensitive isomorphic model [17], leads to greater endogenous complexity. The ongoing investigation [13, 18-20], was distinct from other homeostatic, analogies within Autonomic Computing, including IBM's Horn [14], Kephart and Chess [21], and Herring [22]. Stoyanov [23] managed a Communication Channel to progress viable, Autonomic software systems, prior to modelling maintenance of distributed components [24], whilst Espejo and Hamden [25] cybernetically modelled agent communities. Our research team collaborated with Bustard et al., blending Autonomic Computing, the VSM and Soft Systems Methodology (SSM) [26]. In 2007 we published a mathematical set theory, design grammar model of the relationship between the VSM systems [7], surpassing the Autonomic Computing ideal, towards an original concept of Viable Computer Systems, or VCS. Our 2009 Case Study proved concept in a closed environment scenario [27]. This paper evolves a distinct design grammar model, relating the recursive levels of the VSM, highlighting feedback control within the software process [28, 29] by promoting inter-recursion cohesion to reduce redundancy and so complexity. Context-deviation from human-agency accommodates Autonomic Computing criteria. Inclusion of a temporal element will promote viability in a partially open-bounded environment. The advance design of multi-agent, self-organizing, software systems.

3. TRANSLATION OF VSM INTO VCS CONTEXT

3.1 Viable System Model

Fig. 1, illustrates Beer’s VSM, as a top-down recursive model constituting five-systems, system one, (S1), through to system five, (S5), plus one ancillary sub-system, System Three Star, (S3*). Designed for application to human organizations, unique in its use of variety engineering [8], controlling complexity via homeostatic loops exhibiting feedback control, allowing attenuation of variety from the parent S1, whilst amplifying the variety in terms of its environments. This promotes self-organization and viability as an emergent property, along with self-production, or autopoiesis.

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, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.


Copyright 2004 ACM 1-58113-000-0/00/0004…$5.00.

These resources all rely on coherence between the elements, each part a viable system containing a complete viable system in fractal-like recursion. All parts work with a common purpose in a changing environment, embracing three elements of Management or metasystem, ensuring integration of the operational units, Operation, the locus of recursion, holding the primary activities and Environment the highest recursion of the metasystem, containing external elements of direct relevance to the system. Three key environmental levels include the micro, macro, (or global, in-which these are embedded) and future, forecasting level. The central, vertical spine encompasses four communicative and one intermittent Algedonic, or alarm, channel, transcoding data amongst the system-environment alliance. Each operates in a reactive, feedback controlled mode. Two internal models endorse viability, mapping between the internal capabilities of the system and a changing environment. Unique amongst the federation, S1 promotes viability by notionally linking to the future environment in real time.

3.2 The Viable Computer System VCS

Our research unites Autonomic Computing principles, with established cybernetic concepts and mathematical set theory. A dual perspective model is derived of Beer's VSM, evolving an algebraic design grammar. The core modelling and analysis formalism, it will enable implementation of the proposed VCS specification. Where research has traditionally investigated the requirements of such a system, this study resolves the means to that objective, so advancing the design of a J-Reference Model II as shown in Fig.2. The design grammar model updates the human-oriented VSM to that of a computing context, whereby the operating system undertakes the autonomic role of those biological agents. The syntax will represent a model of the internal systemic capabilities instilling recognition of self and non self, by including a temporal dimension. Viability will significantly, also be retained via recognition of environmental changes. The key complexity sensor, S1 uniquely links directly to the open environment, generating an attenuated model of the required world situation, upon which systemic decisions are made. The VCS aims to observe the activities of the human agents as executors of this function, our research focusing on determining
what the system should sense. This study seeks to both manifest and exploit the inherent VSM fractal-type recursive geometry [30].

depicting the internal processes of each operation, i.e. the relationship between the systems, or subscripts and the recursive levels, or superscripts. The latter makes explicit the potential of this recursivity by replicating feedback control so pivotal to the VSM.

We design a self-governing context-free system pertinent to diverse computing settings, combining atomic elements to meet the complexity reduction ideals of the industry [1, 2].

4. DESIGN GRAMMAR MODEL

The design grammar model is a unique, formal, system-wide, context-free, algebraic set theory representation of Beer's cybernetic VSM. The identities are formulated from dual perspectives: two sets of rules characterizing the relationship between the systems; the subscripts and latterly, a representation of the relationship between the recursive levels; the superscripts. The mathematical analogy is solely a medium to articulate research concepts and ideas. Featuring production rules, a symbol set, and vocabulary including atomic elements of the language, the atomic level, \(S_0\) consists of non decomposable constituents, able to autopoietically generate higher levels. The VSM’s infinite recursivity, post-atomic level, is detailed with no explicit starting point or initial conditions. Three levels of recursion are defined: the lowest (atomic) level, named recursion nought, or \(S_0\), with all higher levels to the penultimate infinite recursion, defined as generic, or \(S_n\). The highest level, \(S_n\), exceptionally, is distinct by its lack of an \(S_2\), terminating autopoiesis from this point and spawning of successive recursions. The design grammar model includes syntax representing each of Beer's five main systems, \(S_1...S_5\), plus a further system nought, or \(S_0\). Atomic breakdown of Beer's model demands separation of \(S_1\) into two parts, \(S_2\) being isolated from the metasystem, yet remaining juxtaposed to \(S_3\) and enclosed by the boundary of the new \(S_1\). These unchanged management and operation units, sited devoid of that \(S_2\), are re-classified as \(S_0\). Inert as a component, in isolation, until joining with its associated \(S_2\), this creates an \(S_1\), the next recursion. This also, correspondingly exhibits Beer's autopoeitic capacity.

The recursion within the syntax both promotes and exhibits stability in the chain of operations, because constant values, by definition, retain their configuration and efficacy as functions are continually executed upon them. An identity will follow its recursive string, promoting self-stabilization, aspirant to developing a VCS satisfying Horn’s self-CHOP benchmark [3], via homeostasis. Where \(S_1\) is equal to \(S_0\) in union with \(S_2\), \(E_1\) will be equal to \(E_0\) in union with \(E_2\). System-environment ecology is vital to sustaining emergence and viability [31] of both design grammar models, the specific environments, being crucial to an identity. Our research has proven concept in a closed environment[27], aspiring a variably open and closed environment. Development of the experience-driven models of the three environmental levels will be continuous, formulating and evaluating systemic capabilities with the lossy data models of scanned environments. \(S_4\) will observe, identify and log the system should sense. This study seeks to both manifest and exploit the inherent VSM fractal-type recursive geometry [30].

The design grammar model includes syntax representing each of Beer's five main systems, \(S_1...S_5\), plus a further system nought, or \(S_0\). Atomic breakdown of Beer's model demands separation of \(S_1\) into two parts, \(S_2\) being isolated from the metasystem, yet remaining juxtaposed to \(S_3\) and enclosed by the boundary of the new \(S_1\). These unchanged management and operation units, sited devoid of that \(S_2\), are re-classified as \(S_0\). Inert as a component, in isolation, until joining with its associated \(S_2\), this creates an \(S_1\), the next recursion. This also, correspondingly exhibits Beer's autopoeitic capacity.

**Figure 2. Proposed model of System Four.**
4.2 VSM Topology Post-Application of Design Grammar Model

We have ascertained characteristics necessary, to enable the context shift from the Beerian human agency to VCS software autonomy, as in Fig. 3.. Incorporating $S_3$, into $S_1$ recognizes that intermittent auditing is obsolete. $S_1$ has been dissected, removing $S_2$ to originate a new system, $S_0$, transposing bottom-up to atomic recursion. Reinstatement of the $S_2$ generates the VCS $S_1$, spawning the next recursion. The highest-level $S_2$ is omitted from $S_0$, terminating recursion. The lossy data compression approach [32] will reflect an homomorphic depiction [17], within $S_4$.

4.3 Design Grammar Model Identities Syntax

Examples of Novel Superscript Relationships

There follow key extracts from the superscript design grammar model, illustrating the relationships between the VCS recursions.

$A$: the set $\{a_1, a_2, a_3, a_4, \ldots\}$ $a_i$ is atomic

$M$: Management unit

$O$: Operation

$E$: Environment

$t$: CurrentTime $| t + 1 $: FutureTime $| t - 1 $: PastTime

System Nought ($S_0$): Comprising of the management and operation unit, atomic level identities between both subscript and superscript models are isomorphic, the VCS will atomically recursively emerge upwards from this point. All higher levels change with the varying perspectives of the model, top level distinct by its lack of $S_2$, recursions only spawning when $S_0$ unites with $S_2$.

$$S_{0, i \leq 7} \rightarrow A$$

$$\left\{ \left( S_{0, i \leq 7} \cup S_{2, i \leq 7} \right) \subseteq S_{1, i \leq 7} \right\} \bigcup S_{n+1, i \leq 7}$$

For $N \geq n > 0$

**System One ($S_1$) Implementation:** The design grammar depicts the $S_1$ composition of a management and operation unit, nesting within an higher parent $S_2$ The atomic level, $S_0$, of the Superscript Design Grammar Model isomorphically maps to the respective $S_0$ within the Subscript Model [7]. $S_1$ is embodied as the union of $S_{0, i \leq 7}$ with $S_{2, i \leq 7}$ interacting directly with environments particular to each $S_1$. $S_1$'s autopoietic properties, spawn recursions, within a range of one to seven $S_1$'s per level. $S_1$ is a superset of all $S_1$'s nested at lower levels at the equivalent range-position. Each system at $S_0$ and $S_1$ levels contain an entire self-governing VCS model. The empty set at atomic level reflects bottom-up emergence.

$$\left\{ S_{1, i \leq 7} \supset \bigcup S_{n-1, i \leq 7} \right\}$$

For $N \geq n > 0$

**System Two ($S_2$) Co-ordination:** Anti-oscillatory and local-regulatory $S_2$ element. A specific $S_2$ per each $S_1$, at nought and generic levels is evident, assisting $S_2$ towards integrative function. The creation of $S_1$, by $S_2$'s unity with a related $S_1$ initiates atomic recursion. The highest level identities are accordingly distinct by the lack of $S_2$, terminating the spawning of further recursions.

$$\left\{ S_{2, i \leq 7} \supset \bigcup S_{n-1, i \leq 7} \right\}$$

For $n < N$

**System Three ($S_3$): Control:** Regulates, optimizes and cohesively stabilizes internal systemic activity. Assisted by $S_2$, $S_3$ gives strategic, overall structure, planning and integrating unified activities of the $S_1$'s. As shown within the syntax, $S_3$ enables concatenation of the fellow metasystemic $S_{3-2-1}$ union and the higher, autonomic, directional, $S_2$ and $S_3$. The identities also express the incorporation of $S_3$ into $S_1$, to reflect the VSM context shift from human multi-agent, intermittent auditing, into that of the computerized capability for constant monitoring by the VCS.

$$\left\{ \left( S_{3, i \leq 7} \supset S_{s_3, i \leq 7} \right) \land \left( S_{3, i \leq 7} \supset \bigcup S_{n-1, i \leq 7} \right) \right\}$$

For $N \geq n > 0$

**System Four ($S_4$), Intelligence:** Enables self-reference and development planning, by embedding a model of the system's internal capabilities, assisted by the temporal perspective. It is unique in communicating directly with all three of the local, future and global environments. Similarly, each $S_4$ links directly to its parent and subordinate counterparts, promoting inter-recursive cohesion and a capacity for redundancy reduction and complexity.

$$\left\{ S_{4, i \leq 7} \rightarrow A \supset \left( S_{n(t+1), i \leq 7} \bigcup S_{n(t), i \leq 7} \right) \right\}$$

$$\supset \left( S_{n-1(t), i \leq 7} \bigcup S_{n-1(t), i \leq 7} \right)$$

For $N \geq n > 0$
5. CONCLUSIONS

Our research has sought to advance Autonomic Computing by exploiting its compatibility with established cybernetic bases of enquiry, chiefly Beer's self-governing, recursive VSM topology. Adopting an algebraic approach to exploit the power of this fractulike recursion, autopoiesis, homeostasis and the fundamental ecological dependence, our research goal of a Viable Computer System, is furthered by progression of the design grammar model. Immediate future refinement will execute a VCS case study within a partially open environment, thereby helping proving the concept.

6. REFERENCES