Digital Ecosystems: Conceptual Optimisation to Manage Complexity, Interoperability and Viability

Dr. T. Love
Dr. T Cooper

Abstract

This paper undertakes a conceptual design optimisation analysis of the structural ontology of digital ecosystems. Digital ecosystems are systems: designed systems. This paper identifies two design guidelines/design primitives for digital ecosystem design. These result from application of two classical well-tested systems analysis approaches to two foundational elements of digital ecosystems:
1. The necessary capabilities of system elements that distinguish digital ecosystem from other digital technologies and are essential to gaining their promised benefits
2. The systemic characteristic of programmatic structure (programming language choice) necessary to placing digital ecosystem designed outcomes in optimal areas of the solution set field to maximise stability, interoperability, flexibility, system health and applicability and minimise technical problems and limitations, and ecosystem and co-related human system pathologies. Incidentally, this also reduces opportunity for proprietary control of the digital ecosystem arena and economy.
The paper does this via the application of Beer's VSM analysis and Ashby's Law of Requisite Variety.
These analyses identify specific problems in digital eco-system design not addressed by current approaches and predict how and why they compromise eco-system viability and reduce effectiveness in systems that attempt to gain the benefits form eco-systemic approaches rather than pre-structured hierarchical or peer-related approaches. It identifies how these problems of system pathology and lack of viability result from missing elements of system structure, software choice and means of standardisation.

Introduction

Computer networks have become increasingly complex. The locally networked mainframe-terminal architectures of the 1960s and 70s made the transition into internationally networked client server architectures of the 90s using the Internet and the World Wide Web. The combination of peer-to-peer networking has enabled individual workstations to be internationally linked in real time to allow individual machines to access the information and spare hardware resources available across the network. In the last 10 years in areas as diverse as business and education, suppliers and consumers are linked in increasingly complex ways through brokered middleware systems of Web services and learning object systems. On the software side, during the 80s, a transition was made from procedural to object-based programming. During the 1990s, increasing use has been made of software agents, particularly beneficial are those capable of autonomously acting across networks. On the human side of computer systems, since the 1990s there has been increasing attention to aligning hardware, software and network systems with real world human systems and organisations, leading to the development of virtual organisations and systems software such as UML for creating code to represent the organisational and information management processes. Since the turn of the millennium, the human aspect of computer-based relations has been enhanced by a focus on social and emotional relationship aspects of human computer interfaces (refs).

Taken together, the above transformations in the computing and networking environments have led to proposals that a particular type of networks could be regarded as a digital ecosystem, and in the case of business environments, a digital business ecosystem. The latter follows naturally from 90s literature relating to business ecology in the study of interactions and development of SMEs.

Some of the key criteria of a digital ecosystem are:

- The elements are networked
- Individual computers consume resources and provide resources (i.e. act as both servers and clients)
- Participants vary in their scale, roles, purposes and expertise
- Participants have differences in needs and the resources they can supply
- There is some autonomous activity in the system (perhaps by autonomous agents or by system-based automated learning)
- The system manages collaboration and competition in such a way as to preserve system integrity and to encourage growth in positive outcomes system-wide.

The underlying presumption of the re-coining of networked information-based interactions as digital ecosystems is that by echoing natural systems, computer systems can gain the benefits perceived to accrue to natural systems, i.e. system stability, system transformation over time, system evolution, improved systemic functioning, improved interaction between digital eco-system members and digital eco-system ecological environment etc

This paper takes the concept of ‘digital eco-system’ and explores its conceptual boundaries via two analytical tools from the field of systems analysis chosen because together they provide:

- A well-tested means of assessing whether digital eco-systems and their elements are capable of being viable
- A means of optimising digital ecosystems in terms of managing complexity and interoperability
- A basis for designing digital eco-systems in terms of identifying the essential properties at element and network levels
- An understanding of the key information pathways digital eco-system elements and digital eco-systems must exchange information with their environments
- An understanding of the most appropriate balance of complexity and standardisation in digital eco-systems and their elements and where this must be located (this in turn defines the type of software/programming environment that is likely to be most effective – or rather, it indicates which are likely to be problematic, why and in what ways)
- A basis for designing digital eco-systems and ecosystem elements

- A means of identifying and predicting digital eco-system pathologies and identifying changes necessary for restoring or creating healthy digital eco-system functioning.
- A basis for conceptually linking concepts of digital eco-systems, digital business digital eco-systems, business engineering, virtual organisation development, real world business practices and real world social and economic development processes.

Justification for these claims will emerge later in the paper. The two analytical tools are Stafford Beer's Viable System Model (VSM) and Ashby's Law of Requisite Variety. A recent (2002) European Union discussion paper regards Beer's VSM as central to understanding digital eco-system development for small to medium enterprises (SMEs) (Nachira, 2002). The two analytical approaches are closely coupled in that VSM concepts draw heavily on Ashby's work, and both are derived from Shannon's early work in communication theory and have cybernetics focus – exactly appropriate for researching and designing digital ecosystems.

This paper will first outline Beer's Viable System Model and Ashby's Law of Requisite Variety. It will then explore how these systems analytical tool applies to digital eco-systems. The penultimate part of the paper describes some key weaknesses and system pathologies resulting from them, including the practical real-world consequences. The concluding section will outline the implications for designing digital ecosystems that are more viable, have improved interoperability, and can minimise problems associated with high complexity both of the digital eco-systems themselves and of their purposes and problem environments.

Viable Systems Model

Stafford Beer's Viable Systems Model (VSM) was has been successfully tested over 45 years across a wide variety of complex real world systemic situations involving people, machines, organisations and computerised systems ranging from Business Process Reengineering (http://www-staff.it.uts.edu.au/~jim/bpt/vsm.html); managing cooperative ventures (ref), information warfare (http://www.systemdynamics.org/conf1999/PAPERS/PLEN1.PDF), to the national economic management of Chile (ref).

Beer's VSM comprises five main sub-systems each of which has to be in place and functioning effectively and appropriately for system viability to occur:

Sub-system 1 (S1): The operational elements of the overall system. These interact directly with the system environment and with each other. These are semi-autonomous units, containing their own local management. These operational units and their local management are how the overall system achieves its ends by interacting with the environment. The VSM is recursive and hence each individual S1 contains another compete and whole VSM if viability is to be maintained.

Sub-system 2 (S2): The coordination sub-system. The role of this subsystem is to integrate the management of S1 units so that they act together for the health of the overall system and avoid prioritising their own interests over the interests of the whole systems. I.e. S2 provides the means of avoiding local-suboptimisation of S1 functioning.

Sub-system 3 (S3): Provides the internal control system for all S1 units. It transforms 'policies' from above (Subsystems 4 and 5) into implementable instructions. There is an additional S3* function by which the functioning of the S3 subsystem is audited by gathering information from S1 sub-systems about their functioning. Typically, this is regular and occasional rather than an ongoing audit

Sub-system 4 (S4): Provides a filtering function in three directions. It filters information from S3 to make it manageable for sub-system S5 (see below). It gathers information (intelligence) from the outside environment (the systems other connection with the outside environment as well as the S1 units) and filters and attenuates it for the use of sub-system S5. In addition, it ensures that decisions and policies from sub-system S5 are transmitted to sub-system S3.

Sub-system 5 (S5): Defines the overall direction and purpose of the eco-system whilst drawing on and integrating the briefings about the *external* environment and *internal* functioning of the system from sub-system S4 to provide policy and strategy. These policies and strategies are disseminated by S4 to the internal integrated management function provided by sub-system S3 that is operationalised by sub-systems S1 with their semi-autonomous management and functions coordinated by sub-system S2.

These five sub-systems have particular informational relationship pathways. These are illustrated in Fig 1below.

Insert Fig 1. VSM diagram HERE

A key aspect of any viable system is that it is recursive. In eco-system terms, this is essential to viability with respect to the larger eco-systems of which the eco-system in focus is a part, the smaller eco-systems of which the eco-system in focus is comprised, and the meso-systemic relationships that the eco-system in focus has with associated eco-systems. In terms of Beer's VSM, all S1 sub-systems must be VSMs in their own right and S5 of the eco-system in focus must effectively represent all and every attribute of the eco-system that it guides.

Viability, stability, eco-system effectiveness and interoperability depend on all five sub-systems and the relationships being in place and in appropriate balance. Where sub-systems and relationships are missing, weak or overbearing, then the eco-system as a whole will malfunction pathologically. The generic nature of the morphological representation of the VSM across all and any viable system (whether living, inanimate or informational) means that characteristic eco-system pathologies can be identified and the style of failure outcomes predicted. Conversely, observed system failings fall into pathological categories that indicate problems with specific sub-systems and their relationships. These relations between weaknesses in system structure as defined by the VSM and specific systemic pathologies and lack of viability have been consistently identified across a very wide range of system types, and it is unlikely that digital eco-systems would be any different. In fact, the rational simplicity and informatic basis of digital ecosystems would be expected to indicate more clearly these pathologies as compared to the biological complexity of eco-systems with a higher ration of organic/biological involvement.

An important issue is that the VSM sub-systems S1-S5 and their necessary relationships are a homomorphic representation of system functions in a viable system. In most cases, they do not map directly onto real world system elements. For example, the CEO of a business is not subsystem S5. The CEO may or may not undertake some or all of S5 subsystem functions, or they may be done elsewhere in the organisation or not done at all. Separating the essential functioning from the real-world forms is one way that the VSM offers diagnostic benefit in identifying what needs to be done to maximise the viability, stability and interoperability of systems such as organisations and digital eco-systems.

Ashby's Law of Requisite Variety

William Ross Ashby was a psychiatrist involved in the earliest stages of the study of complex systems whose work influenced many in that field to the present. His law of requisite variety is stated in many different ways such as 'only variety can absorb/control variety' or ' ' (refs). In essence, his law of requisite variety states that to control a complex system requires that the controlling subsystem must be capable of a similar variety of states as the system itself ('Every Good Regulator of a System must be a model of that System itself' (ref).

For complex, layered and hierarchical systems, the variety generating elements and the controlling subsystems may be distributed at many points in the system. Overall, however, Ashby's Law of Requisite Variety is still expected to hold in that for the system to be controlled (i.e. stabilised) the variety of control available to the controlling subsystems must be at least equal to the variety of states possible for the uncontrolled system.

Significantly, for complex, layered and hierarchical systems, the style of outcome in terms of stability depends on the locations of the subsystems generating variety and the controlling subsystems able to use variety of responses to control system variety. Alternatively, where differing sub-systems of control are involved in the management of a system and some sources of control are able to increase their variety to accommodate the lack of requisite variety in other control systems then the style of control will be shaped by the amount of transfer to the accommodating control system. This is an extension, by the author, of Ashby's Law. It is a significant in:

- Designed systems which are under development
- Situations involving rework
- Evolving social situations
- Situations to which standards apply but which do not completely define solutions
- Complex, evolving, autonomous and semi-autonomous systems

For example, a design team of a new motor car will apply what they perceive to be the requisite variety to control the production of the design for a vehicle that is safe, can be manufactured as specified and function as intended. This includes design process, design checking, engineering and market research etc. Additional variety-attenuation and requisite variety-assessing methods (such as prototyping, user testing) may be used to ensure this outcome. Any outstanding variety relating to the vehicle, however, will be accommodated through alternative control mechanisms such as in-production design modifications, rework repairs, product recalls, and product development modifications (usually incorporated into a new version of the vehicle). These latter methods mop up excess variety in possible system states uncontrolled by the requisite variety offered in the design stages.

The more variety is controlled in the earlier stages of the product development the more the product is similar to what was conceived and intended. As the variety exceeds what is matched by the requisite variety of the control systems, then if the outcome is to be controlled, it must be done so by the applications using additional variety later. Often, in practical situations, this later application of control is ad-hoc, inefficient and has knock-on adverse outcomes.

Digital eco-systems and VSM

The success of Beer's VSM as a homomorphic representation of the minimal condition for system viability over a wide range of human, mechanical, social, digital, organisational and networked systems suggests that it is appropriate to use as a basis for reviewing digital eco-systems and, perhaps, for developing digital ecosystem design parameters.

Digital-ecosystems comprise a variety of sub-systems using hardware, software and human intervention. A key intention of digital eco-systems is to use the protection of the eco-system via managed subsystem collaboration improve eco-system outcomes and improve outcomes for individual subsystems. To do this requires that the digital eco-system is viable and does not have systemic pathological traits.

Beer's analyses suggest that in using VSM to analyse the viability and pathology of digital ecosystems it is necessary to make sure that the necessary subsystem functionalities and informatic relationships are in place for three digital eco-system recursions:

- The completeness of the VSM of the digital ecosystem that is of direct interest (DEmain).
- The viability of the VSM of subsystems S1 of DEmain. These are the operational subsystems by which DEmain creates value-producing interactions with its external environment.
- The ability of subsystem S5 of DEmain wholly to represent DEmain (as a complete viable VSM) to a larger viable system of which DEmain is an S1 subsystem.

A simple table and tick list provides sufficient analytical technology for the purpose.

Table 1: The completeness of the VSM of DEmain

Subsystem element	**VSM role**	**DEmain processes fulfilling this functionality**	**Completeness of function**	**Comments**
S1				
S2				
S3				
S3*				
S4				

S5				

Digital eco-systems typically comprise fully automated sub-systems, semi-autonomous sub-systems, human-controlled subsystems, human organisations, humans and, sometimes, other organic subsystems (e.g. in digital ecosystems relating to food production).

Most system designers recognise the need for sub-systems S1, S3 and S5. In real-world human organisations, these roughly correspond to the people that do the work (S1); the people that manage the people that do the work (S3); and the (S5) people whose idea it is to have that organisation with that particular purpose and who direct the S3 managers. The literature over 45 years of experience in using Beer's VSM to diagnose organisational pathologies indicates, however, that the S2 coordination functions and the S4 intelligence gathering and filtering functions are often neglected. This neglect of S2 and S4 leads to operational inefficiencies and control of the organisation being forced by S1 units who are managing blind (no intelligence) and aiming to improve their lot at the expense of the eco-system as a whole. It also typically leads to collapse of the S5 and S3 functions together into strategy-less reactive management.

Subsystem S1 of DEmain can be similarly assessed in terms of its system viability and pathology:

Table 2: The completeness of the VSM of S1 of DEmain

Subsystem element	**VSM role**	**DEmain processes fulfilling this functionality**	**Completeness of function**	**Comments**
S1				
S2				
S3				
S3*				
S4				
S5				

Assessment of the satisfactoriness of sub-system S5 of DEmain to represent DEmain as a whole viable eco-system to the eco-system within which it is subsumed as an S1 sub-system is provided by Table 1 and Table 2. The potential for a pathological DEmain to compromise an eco-system within which it is subsumed is limited. In the main, it compromises the functionality provided by itself and an S1 subsystem. It may also, however, compromise higher functioning by making subsystem S2 coordination of other operational subsystems difficult. A seriously rogue S1 may also result in resulting in additional resources being used in internal management by S3 subsystems and feedback loops, by S3* auditing functions, by S4 sub-system in its attempts to balance internal and external situations, and by S5 in creating policies and strategies that attempt to create a viable eco-system whilst managing the effects of the rogue S1. A real-world example of this situation is when a business is bought by another business, and the buying firm links the two firms together but retains the management of the bought firm and provides them with full authority to manage the bought in business. The potential is there for a reverse takeover (reference).

Digital Eco-systems and Requisite Variety

Ashby's Law of Requisite Variety and the corollaries and extensions developed by others offer significant insights into managing and designing digital eco-systems. Changes to the distributions of environmental, open system and controlling variety in a digital eco-system changes the distribution of the different loci of control of participating eco-system elements and organisations, including those who provide internal information flow management services (e.g. network services, middleware, database management services, brokerage and coordination , access, authorisation infrastructure, financial management etc).

The digital eco-system example below shows how the distribution of power and control hegemonies between players, and even the structure and form of the digital eco-system is significantly dependent on something as simple as standardisation of underlying software – through its role in changing the distribution of variety in the digital eco-system.

Example Digital Eco-system: Learning Object Economy

On example of a digital eco-system is that of learning object education systems. Learning object systems are based on the idea of reusable learning content packaged and served digitally as 'learning objects' that can be combined in different ways with other learning objects for different teaching and learning situations (Alvarado-Boyd, 2003). Typically, learning object systems depend on learning objects being attached to pre-defined 'learning object meta-data' (LOM) that can be indexed and queried by a learning object management system (LMS). The digital ecosystem comprises multiple players both providing resources (learning content converted into learning objects) and accessing the resources of others. The transfer of learning object resources is undertaken under a variety of economic mechanisms, e.g. some free, some proprietary, some pre-paid and some bought on demand.

The meta-data is usually incorporated into, or wrapped around, learning content objects using mark-up languages (e.g. . are meta-data for html web pages using Title, Keyword and Meta Tags).At a larger system level, learning object meta-data insertion and management has focused on XML-based approaches in which learning objects are located in learning content databases. A limitation of this mark-up-based approach using html/XML is that this type of digital ecosystem is restricted to virtual or digital learning objects that can be stored digitally with their meta-data and be network accessible.

Systems analysis of this digital ecosystem of learning object systems via Ashby's Law of Requite Variety and sundry extensions and corollaries indicates that there are significant systemic structural problems with the use of XML and other mark-up languages for codifying learning object meta-data that merge as poor interoperability at all system levels, system inefficiencies, poor eco-system viability, need for additional supporting system structures and problems of hegemonic control of the whole digital eco-system by propriety interests that supply systems to repair problems of system viability.

Broadly, the problem relates to focusing attempts at building system standardization and interoperability of meta-data and data by the use of XML at the lower system levels, typically at page-description level. This contrasts with alternative approaches such as RDF that focus on standardization at the level of over-arching system framework.

Where standardization attempts are focused on the lower levels of the system, improving interoperability between units, courses, servers, networks and institutions requires strategies that difficult to implement and structurally add to the overall problem. Coding meta-data via inline XML and html mark-up requires learning object content and page elements have meta-data applied in consistent pre-defined and pre-structured ways to be consistently machine-readable. This depends on pre-specified, pre-defined and accurately applied meta-data vocabularies. In addition, managing interoperability between different LMSs, different networks and different institutions has required ongoing development of multiple middleware, database and communication standards many of which are proprietary. All of these are problematic in a digital learning content eco-system whose meta-data classification is emerging as time passes and where learning objects are classified by meta-data in a variety of different ways.

Addressing the problem of interoperability is complex because it involves a variety of technical, human, conceptual, educational and informatic considerations. Systems analysis tools such as Ashby's Law of Requisite Variety (see, for example, http://pespmc1.vub.ac.be/ASC/LAW_VARIE.html and http://www.cybsoc.org/ross.htm) is of particular relevance to analysing the current trajectory of learning object eco-system development. The Law of Requisite Variety predicts that to be satisfactory, the complexity and scope of all aspects of digital learning objects eco-system management must echo the variety in human learning, the variety in learning object data and meta-data, and the variety in the contexts in which the learning object systems, users and creators' function and interact.

From the perspective of Ashby's Law of Requisite Variety, it is problematic to attempt to propagate standardization (i.e. achieve control by attenuating system variety upward from the page content level via mark-up languages when variability occurs at many levels and in ways that are not addressed by page level standardization. This attempt at control is essentially 'back to front'. Attenuating all eco-system variety by control implemented by what is primarily a page-description standardisation method is insufficient for viability of an interoperable digital learning object eco-system because it does not fully control the variety across and between different learning objects, courses, learning designs, software systems, disciplines, organizations, networks, and other technical, virtual and real institutions.

From a superficial perspective, the transition to XML after html appeared an improvement (it increases control variety). At a whole system level, the gains offered by using XML and XHTML are limited. XML was designed for commercial systems with already tightly controlled variety of object types are transacted in limited ways with transactions undertaken close to page level. In contrast, digital learning object eco-systems are high variety systems.

The limitations of XML to attenuate system variety, however, requires additional sub-systems to control variety at higher levels in the system, for example, variety due to differing uses of the same object, differing higher level classification systems, different forms of machine parsing engines with different parsing approaches, differing learning object data structures in different organizations and different computer systems, and even differing interpretations of the XML standards. This is the focus of current intensive and expensive effort by e.g. ADI, IMS, IEEE, and OSPI to create multiple mid-level standards such as SCORM. Whilst being ways to attenuate existing system variety, these add to the problem by increasing variety overall that needs to be absorbed. That is, attempts to resolve the structural problems by using XML plus additional systems tends to result in increased variety, increased complexity, weaker interoperability and increased dependence on incompatible proprietary formats each addressing different problematic aspects of standardization that will in turn require more standardising subsystems until eventually the complexity of the standards and variety controlling systems matches the variety in the whole system. This is potentially an unconstrained problem because the system is effectively unrestrained at upper systems levels.

That is, the underlying structural weaknesses of XML-based learning object systems will continue to produce problems of incompatibility between systems, continue the problems of lack of flexibility in responding to change and to new insights, and the lack of scalability in processes. These problems of poor variety attenuation of XML and mark-up languages are substantially resolved by the Semantic Web infrastructure such as the Resource Description Framework (RDF).

Using RDF as the basis for a digital eco-system management offers improvements in eco-system viability and reductions in eco-system pathologies compared to XML. RDF controls variety from the top down. RDF specifies standards and interoperability at network level, and as a framework for interoperability propagates standardization via simple graph-based 'triple' protocols downwards to the page level, where it can be efficiently actualized via, RDF/XML an RDF-based variant of XML that integrates well with existing XML page descriptions. It attenuates system variety whilst offering better control variety by increasing the variety of the system and its communication channels. This enables the management of high levels of variety in software systems at page level.

In practical terms, RDF allows separation of metadata describing learning objects from the objects themselves. It allows the integration of different forms of meta-data; provides a smooth transition to consistent vocabularies as and when they are available and appropriate, provides graceful resolution of inconsistent meta-data and relative avoidance of incompatible meta-data.

Conclusions

This paper has outlined some of the analyses from design6focused research aimed at identifying key design factors affecting the viability, interoperability, system effectiveness and efficiency, hegemonic control of system development and distribution of economic gain by real world proprietary constituencies.

The paper outlined the application of Ashby's Law of Requisite variety and corollaries to analysing structural ontology of digital eco6systems to identify systemic constraints on successful designs for digital eco-systems and their components. Additionally, the paper reports an extension of Ashby's Law of requisite variety that relates relative distribution of system variety and control variety to opportunities for hegemonic

control of system development and distribution/acquisition of economic and other value from the system.
The paper demonstrated this via

References

Alvarado-Boyd, S. (2003). *Traveller's Guide to the Learning Object Landscape*. Austin: New Media Consortium.

Koper, R. (2004). Use of the Semantic Web to Solve some Basic Problems in Education: Increase Flexible, Distributed Lifelong Learning, Decrease Teachers' Workload. *Journal of Interactive Media in Education, 2004*(6), 1-9.

Nachira, F. (2002). *Towards a Network of Digital Business Ecosystems Fostering the Local Development*. Brussels: European Union.

4570 words