On the Complexity of Engineered Systems (and its effect on protocol deployment)

David Meyer

dmm@1-4-5.net

Abstract

While advanced technological and biological systems have very different implementations at the molecular and component levels, they show surprising similarities in systems-level organization. In fact, Systems Biology can inform network engineering as to system-level architectural features that imbue advanced engineered systems with scalability, adaptability and evolvability. Perhaps most surprisingly, convergent evolution in both domains has produced modular architectures that are composed of elaborate hierarchies of protocols and layers of feedback regulation in response to the demand for robustness to uncertain environments. These somewhat paradoxical features are neither accidental nor artificial. Rather, they derive from a deep and necessary interplay between complexity and robustness, modularity, feedback, and fragility. This paper explores insights from both biological systems and engineering theory that can help inform the design and deployment of engineered systems such as the Internet as well as helping to describe why some technologies are difficult to evolve and deploy.

1 Introduction

While advanced technological and biological systems have very different implementations at the molecular and component levels, they show surprising similarities in systems-level organization. In fact, Systems Biology [3] can inform network engineering as to system-level architectural features that imbue advanced engineered systems with scalability, adaptability and evolvability. Perhaps most surprisingly, convergent evolution in both domains has produced modular architectures that are composed of elaborate hierarchies of protocols and layers of feedback regulation in response to the demand for robustness to uncertain environments. These somewhat paradoxical features are neither accidental nor artificial. Rather, they derive from a deep and necessary interplay between complexity and robustness, modularity, feedback, and fragility. This paper explores insights from both biological systems and engineering theory that can help inform the design and deployment of engineered systems such as the Internet as well as helping to describe why some technologies are difficult to evolve and deploy.

This paper briefly explores principles learned in the biological domain with an eye towards gaining a deeper understanding of the implications of architecture on the design and deployment of Internet technologies. One perhaps surprising consequence is that while "narrow waist" architectures[1] which provide for diversity and evolvability above and below the waist, it is just this property that at the same time causes difficulty in deploying new technologies in the waist itself. This effect can be seen in biological systems as well as technological systems and perhaps surprisingly, is an inherent feature of robust and evolvable systems.

The remainder of this paper is organized as follows: Section 2 briefly introduces the ideas underlying Robustness and Fragility. Section 4 discusses system-level architectural features for scalability and evolvablity, and discusses how these same features inhibit the deployment of protocols with specific features. Section 3 provides one definition of complexity, and Section 5 provides a view to the future as well as recommendations as to what we, the Internet engineering community, should consider as paradigms for more deeply understanding the network. Finally, Section 6 provides a summary and *More Questions Than Answers*.

2 Robustness and Fragility

Figure 1 depicts a typical "Complexity-Robustness" curve for a class of familiar systems which includes the Internet and many biological systems [4]. As we can see such systems require some degree of complexity to acquire robustness. However, beyond a certain point (P_{max}) additional complexity causes fragility and brittleness. While detailed discussion of robustness, fragility, and complexity are beyond the scope of this paper, the next sections provide an overview of these topics.

2.1 Robustness

Definition: A *[property]* of a *[system]* is robust if it is *[invariant]* with respect to a *[set of perturbations]*, up to some limit [2].

Another way to think about this is that robustness is the preservation of a certain property in the presence of uncertainty in a system's components or environment. Interestingly, a system can have the property that it is robust to one set of perturbations and yet fragile for a different property and/or set of perturbations. Such a system is called *Robust Yet Fragile* (or RYF-complex). For example, a possible *RYF tradeoff* is that a system with high efficiency (i.e., using minimal system resources) might be unreliable (i.e., fragile to component failure) or hard to evolve. Examples of systems exhibiting RYF-complex behavior include many biological signaling paths and the Internet [2].

[1] Such as the Internet architecture

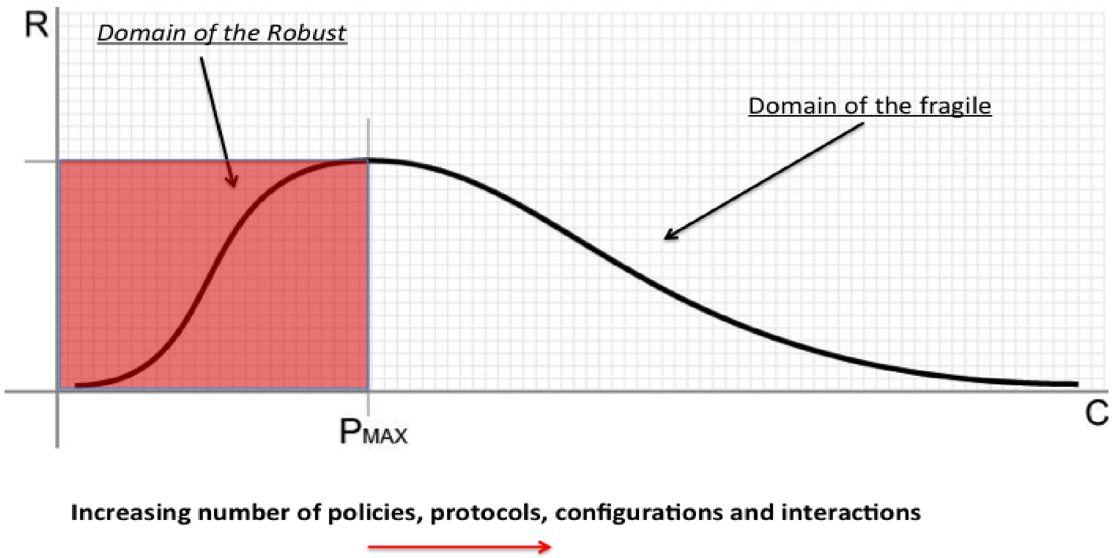

Figure 1: Robustness vs. Complexity – Systems View

Note that we can characterize many familiar system properties as forms of robustness. For example:

- **Reliability** is robustness to component failures
- **Efficiency** is robustness to resource scarcity
- **Scalability** is robustness to changes to the size/complexity of the system as a whole
- **Modularity** is robustness to structure component rearrangements
- **Evolvability** is robustness of lineages to changes on long time scales

2.2 Fragility

Fragility can be most easily seen in an example. Consider a coffee cup on a sitting on a table. The cup is *fragile* because suffers non-linearly more from large deviations than from the cumulative effect of smaller events. More concretely, the cup is dropped on average 1 cm every time it is set down. Picking it up and setting it down say, 300 times, has no effect on the cup. However, dropping the cup from 3 meters destroys it (i.e., the damage grows in a non-linear fashion). Another example: consider the difference in damage to a human who jumps off something that is, 0.3 meters high 30 times versus jumping off something that 9 meters high one time; in the first case, no real damage; in the second case the person is dead.

In the network world, we see several examples of fragility, including

- ARP storms [15]
- Micro-loops in the Internet routing system [14]
- TCP congestion collapse [10]
- The AS 7007 route leakage incident [17]

2.2.1 Formal Definition of Fragility

Let z be some stress level, p some property, and let $H(p,z)$ be the (negative valued) harm function Then for the fragile the following must hold

$$H(p, nz) < nH(p, z) \text{ for } 0 < nz < K \quad (1)$$

where K is the level at which the system collapses. We say such a system is *K-fragile*. The intuition here is that the cumulative damage to a fragile system from a series of small shocks is non-linearly less that the damage from one large shock [16].

Note that Equation 1 is a variation on Jensens Inequality [7] and importantly is not mean-preserving for convex H. In particular

$$H(p, (z1 + z2)/2) \neq (H(p, z1) + H(p, z2))/2 \quad (2)$$

That is, *the convex function of the mean is not equal to the mean of convex functions*. Assuming that a distribution's mean is preserved by averaging is a common technique when the underlying distribution is unknown (perhaps the most common case). As shown here however, such averaging leads to model error and hence additional uncertainty.

Finally, there is a close relationship between fragility as defined here and what we typically think of as scaling. For example, when we say something scales like $O(n^2)$ what we mean is the damage to the network has constant acceleration (2), and as such is of concern as n gets large.

3 Defining Complexity

Alderson and Doyle [2] provide the following insightful description of complexity:

> *In our view, however, complexity is most succinctly discussed in terms of functionality and its robustness. Specifically, we argue that complexity in highly organized systems arises primarily from design strategies intended to create robustness to uncertainty in their environments and component parts.*

That is, complexity is really about *structure* that is "designed" to create robustness to environmental and component uncertainty. We will return to this theme later in this paper.

4 Architectural Structures for Scalability and Evolvability

In this section we describe six *systems-level architectural features* found in both technological and biological systems that confer scalability and evolvability. These include bowties (thin waists), protocols, massively distributed, *robust* control[2], a high degree of layering, and component reuse. Interestingly, these are also the same features that result in the RYF-complex behavior of these systems[3]

4.0.2 Bowties and Thin Waist Architectures

While the view of architecture as *constraints that deconstrain* originated in biology [11], similar (if not isomorphic) concepts are found in many engineered systems, including the Internet[8] [1] and many manufacturing, transportation and economic systems[9]. In biology, this structure is typically called a bowtie. In engineered systems like the Internet, the same structure is generally called an hourglass (the key difference revolves around whether a system's layering is seen as either horizontal or vertical). A robust architecture is constrained by protocols, but the resulting plug and play modularity that these shared constraints enable deconstrain (i.e., make flexible) systems designed using this architecture. Constraints give a convenient starting language to formalize and quantify architecture and ultimately, a mathematical foundation[4]. In the domain of manufacturing (in this case clothing), consider a given wardrobe that is a collection of garments and the problem of assembling an outfit that provides suitable robustness to the wearer's environment. Three distinct but interrelated types of constraints are universal in clothing as in all architecture: (i) component (garment) constraints, (ii) system (outfit) constraints, and (iii) protocol constraints. Therefore, in combination, diverse, heterogeneous components (garments) that

[2] Here the term "robust" is used in the control-theoretic sense, i.e., risk aware controllers such as TCP/IP

[3] Note that it has been shown that RYF-complexity is conserved so while this behavior might be mitigated (to some degree), it cannot be eliminated [6].

are constrained by materials and construction combine synergistically (through protocols) to yield outfits that satisfy system constraints not directly provided by any single component. Note that these system constraints are *weakly emergent*[5].

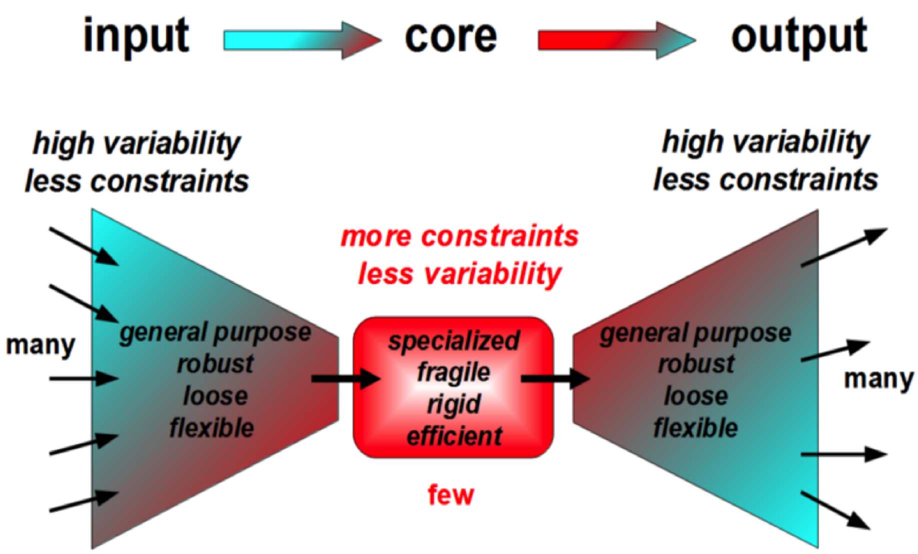

Figure 2: Biological bowtie

Figure 2 shows a typical biological bowtie. Here the focus is on the set of the metabolic reactions and processes that take place in the cells of organisms to convert biochemical energy from nutrients into adenosine triphosphate (ATP). ATP is the "knot" of the bowtie, as every cell uses ATP for energy transport (think of the ATP molecule as being analogous to an IP packet). Many components can be used to make ATP, and all of life uses ATP for energy transport. Note that if you turn this bowtie on its side see something that looks like the classical "Internet Hourglass (Figure 3).

As shown in Figure /reffig:comparison, the bowtie and hourglass are similar architectural features who's difference lies mainly in whether one views layering as horizontal or vertical.

In summary, bowties are a fundamental systems-level structure found in both biological and technological systems that are scalable and evolvable. *However, the same structure that allows for the remarkable diversity and robustness of life on earth (or the Internet) is the same structure that makes, for example, IPv6 hard to deploy.*

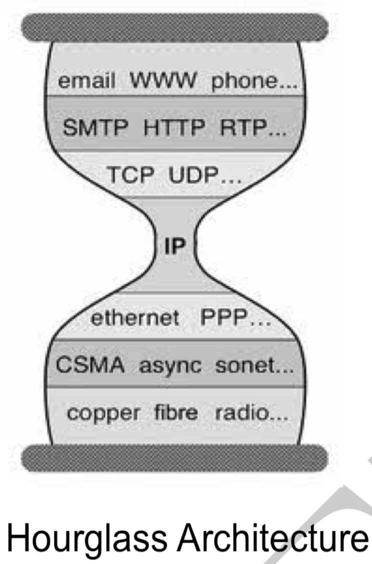

Figure 3: Internet Hourglass

5 A View to the Future

The underlying thesis embodied in this paper is that complex systems such as the Internet can be understood only by identifying their organizing principles, theories, design rules, and protocols. Hence the approach advocated here is more akin to Systems Biology in that it seeks to find a complete computational system model which embodies such organizing principles. These principles lead us to understand why deployment of certain protocols, most notably those that exist around the "bowtie" are hard to change. We have seen this effect embodied in the difficulties in deploying with both IPv6 and DNSSec.

In summary, the claim articulated here is that as an engineering community, we need to develop a multidisciplinary approach to describing, understanding, and controlling the properties and dynamics of the whole network. Of course, this requires the integration of information from many sources. Fortunately, there are highly organized (and common) universal structures (architectural features) underlying biological and technological networks. These are the structures mediate that effective tradeoffs among efficiency, robustness, and evolvability with predicable fragilities while at the same time enforcing structure that makes acceptance and deployment of certain technologies difficult. Unfortunately, the theory for the type of distributed and asynchronous global control used in the Internet (or biology) is relatively new (and its natural language is mathematics). That said, it is still possible to find insight into candidate universal structures that can be identified by comparing biological and technological systems [12].

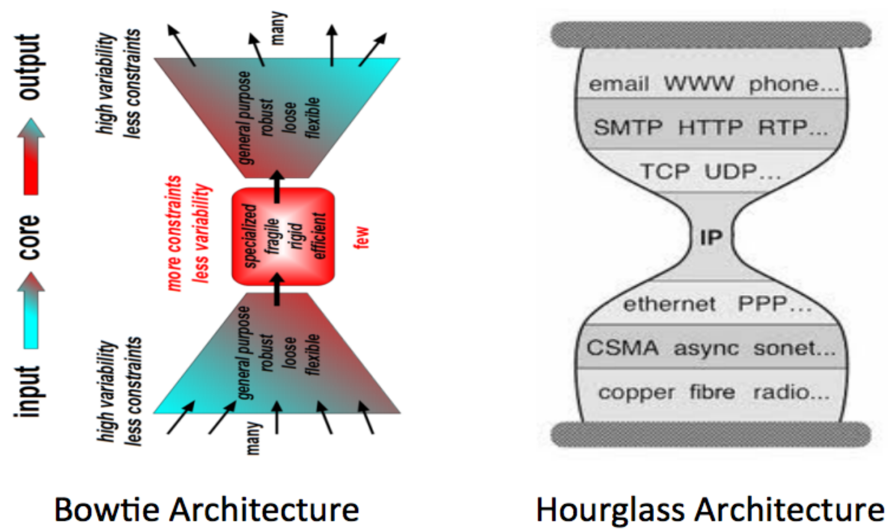

Figure 4: Comparing the Bowtie and Hourglass

6 More Questions than Answers

While we are just at the beginning of what is undoubtably a long endeavour, we have learned that there are fundamental architectural building blocks found in systems that scale and are evolvable. These include

- RYF complexity/tradeoffs (really a behavior)
- Bowtie architectures
- Protocol Based Architectures (PBAs)
- Massively distributed with robust control loops
- Complexity-robustness spirals
- Highly layered

However, while these structures lead to scalability and evolvablity, the "knot" of the bowtie is notoriously unevolvable (again, this follows for the notion of *constraints that deconstrain*). This is, at least in part, what make protocols that live in the "knot" hard to evolve (read: change). While the idea of narrow waists (or bowties) is well socialized in both the networking and systems biology communities, deeper understanding of how these architectural features inhibit evolvability of protocols and modules that make up the knot

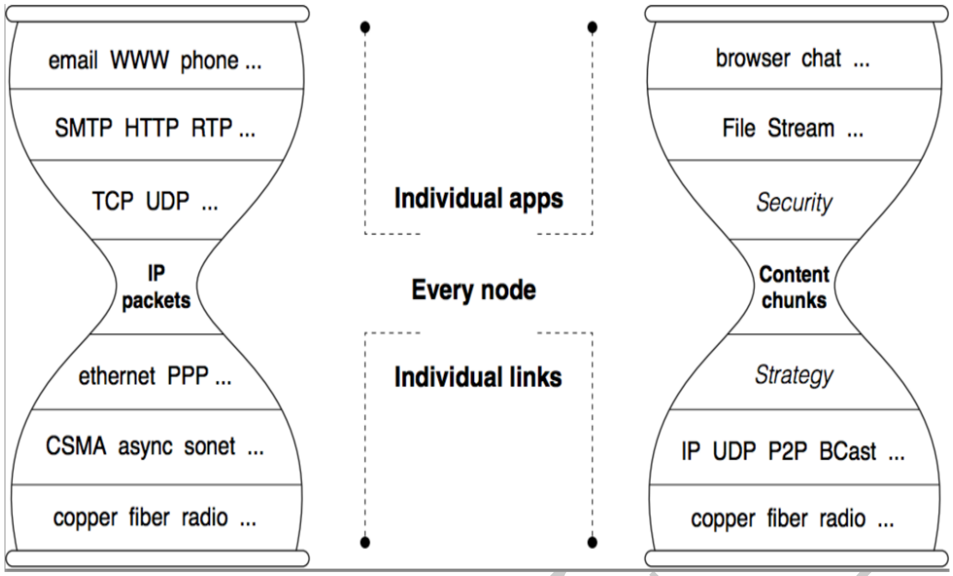

Figure 5: Classic Internet and NDN Hourglasses

is an area of active research. For example, Figure 5 depicts an hourglass architecture for Named Data Networking [13] and compares it to the classic Internet hourglass. Finally as noted above, while we focused on the structure of the "bowtie" in this brief discussion, the bowtie is only one of the architectural features which simultaneously confer scalability and evolvability while at the same time making protocols in the "knot" difficult to evolve. Understanding these structures and how they interact with other socio-economic factors is fundamental to our understanding of deployability of new protocols in the Internet.

References

[1] Saamer Akhshabi and Constantine Dovrolis. The evolution of layered protocol stacks leads to an hourglass-shaped architecture. In *SIGCOMM*, 2011.

[2] D. Alderson and J. Doyle. Contrasting views of complexity and their implications for network-centric infrastructures. In *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Human*, volume 40 Issue 4, pages 839– 852, 2012.

[3] Frank J. Bruggeman and Hans V. Westerhoff. The nature of systems biology. *TRENDS in Microbiology*, 15(1), 2006.

[4] J. Carlson". Highly optimized tolerance:a mechanism for power laws in designed systems. *Physical Review E*, 60(2):1412–1427, 1999.

[5] David J. Chalmers. Strong and weak emergence. *Australian National University*, 2006.

[6] Marie E. Csete and John C. Doyle. Reverse engineering of biological complexity. *Science*, 295, 2002.

[7] I Csiszar. A note on jensen's inequality. *Studia Sci. Math. Hungar*, 1, 1966.

[8] Steve Deering. Watching the waist of the internet hourglass. *IETF*, 2001.

[9] John C. Doyle and Marie Csete. Architecture, constraints, and behavior. In Donald W. Pfaff, editor, *Proceedings of the National Academy of Sciences of the United States*, volume 108, pages 15624–15630, Septemeber 2011.

[10] Sally Floyd and Kevin Fall. Promoting the use of end-to-end congestion control in the internet. *IEEE/ACM Trans. Netw.*, 7(4):458–472, August 1999.

[11] Marc Kirschner and John Gerhart. Evolvablity. In *Proceedings of the National Academy of Sciences of the United States*, volume 95, pages 8420–8427, July 1998.

[12] David Meyer. Macro trends, complexity, and software defined networking, http://www.1-4-5.net/ dmm/talks/ausnog2013.pdf, Septemeber 2013.

[13] Named Data Networking. http://named-data.net.

[14] M. Shand and S. Bryant. IP Fast Reroute Framework. RFC 5714 (Informational), January 2010.

[15] S.Vidya and R.Bhaskaran. Arp storm detection and prevention measures. In *IJCSI International Journal of Computer Science Issues*, volume 8, March 2011.

[16] Nassim Nicholas Taleb. Black swans and the domains of statistics, 2007.

[17] Xiaoliang Zhao, Dan Pei, Lan Wang, Dan Massey, Allison Mankin, S. Felix Wu, and Lixia Zhang. An analysis of bgp multiple origin as (moas) conflicts. In *Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement*, IMW '01, pages 31–35, New York, NY, USA, 2001. ACM.