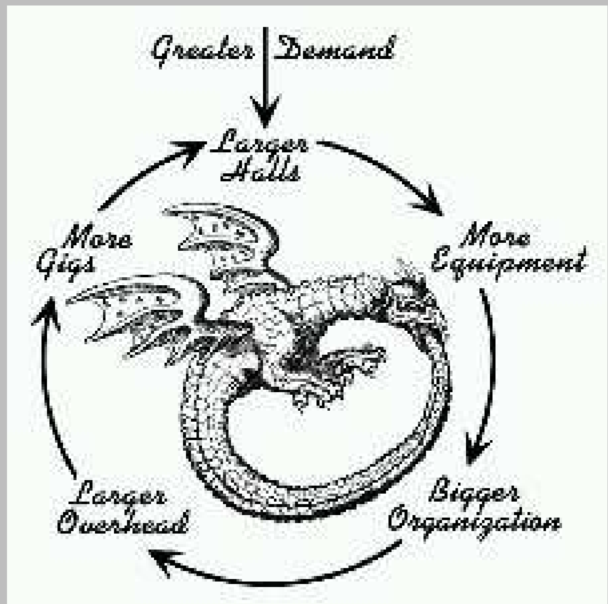
Back to the Future
Complexity, Convergence, Economics, Public Policy and the Future of the Internet



David Meyer
13 Apr 2006
http://www.1-4-5.net/~dmm/talks/back_to_the_future

Outline

- Introduction
- A Few Observations
- Convergence?
  - Geoff Huston
- Q & A

Introduction

- So what is this event about?
  - Well, we all want to hear Geoff’s talk
- But also...the growing complexity of the Internet technology, and its implications
  - On what we build
  - On the economics of our industry
  - And on public policy
  - And on the possible future structure of our industry
  - What is interesting is all of the factors have started to come together
- Goal: Socialize the understanding of these factors within Cisco

So...I'll Start with a Few Observations

- On complexity
  - What is it, how does it arise, what can/should we do about it?

- On how complexity interacts with our technology

- In particular, how complexity interacts with the Internet Architecture and
  - The economics of (IP) packet carriage, and
  - Public policy

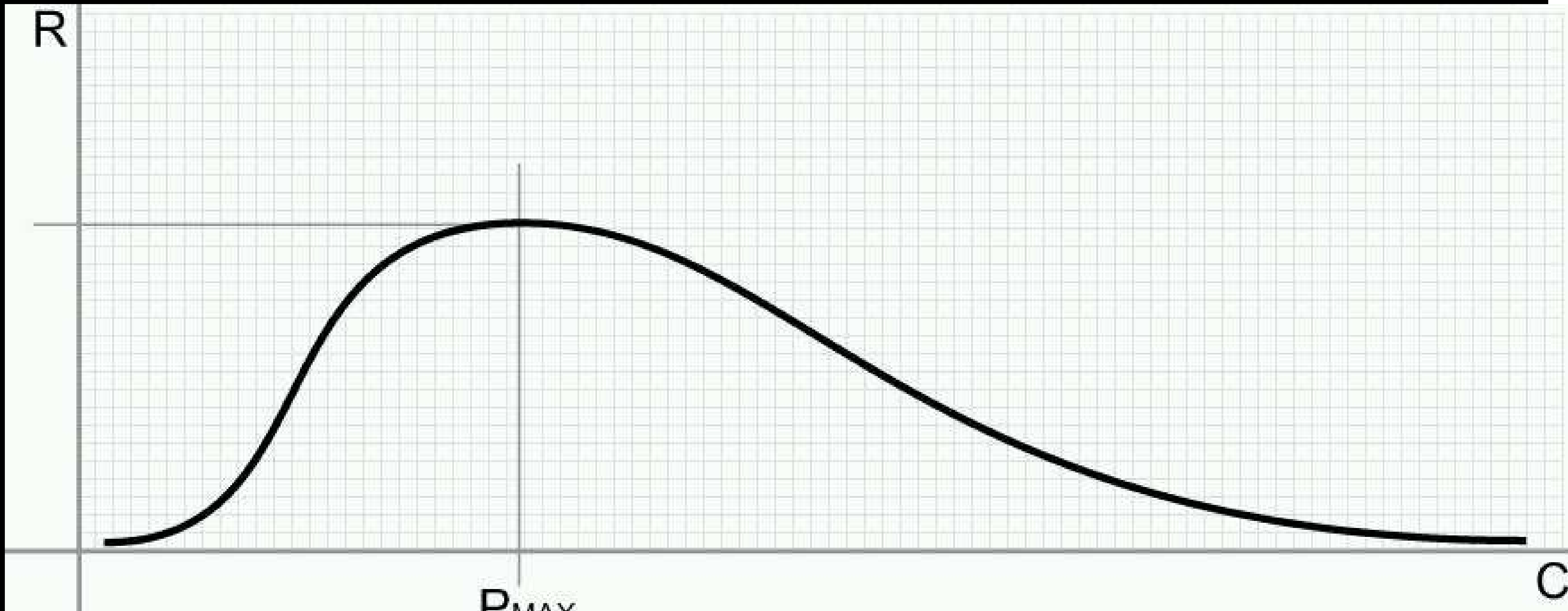
So, complexity...

- First, realize that complexity of the sort I'm going to be talking about is not well understood/hard to quantify
  - We just don't have the tools (yet)
- Our focus here is on the non-linearity inherent in large systems such as the Internet
- Recent work in the dynamical and non-linear systems field suggests thata class of highly designed systems (such as the Internet) are "Robust yet Fragile"
  - Sounds like a contradiction of sorts, doesn't it?

Robust yet Fragile Systems?

- Basically, a system needs some degree of complexity (design) to achieve robustness
  - But at the cost of increasing sensitivity to (hopefully) rare events
- However, while a system's robustness is bounded, its complexity is not
- To be a little more quantitative, the complexity-robustness curve is heavy-tailed
- Let's take a look at what this means

What does all of this mean?



A system needs some degree of complexity to achieve robustness

See "Complexity and Robustness", J.M Carlson and John Doyle, Proc. Natl. Acad. Sci. USA, Vol. 99, Suppl. 1, 2538-2545, February 19, 2002. http://www.pnas.org/cgi/doi/10.1073/pnas.012582499 for a nice introduction.

Sources of Complexity

- There are also two well-known principles from non-linear systems theory that play into this space:
- Amplification
- Coupling

Amplification Principle

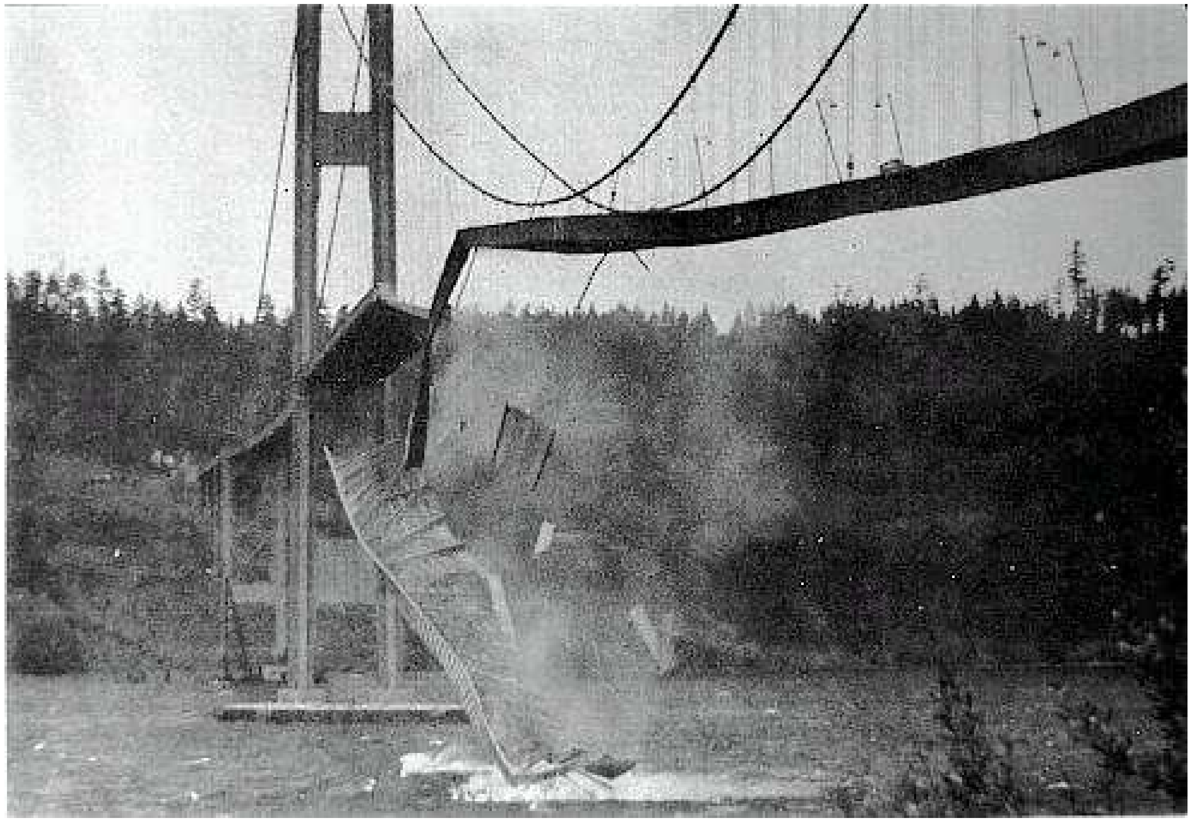
- The Amplification Principle states that there are non-linearities which occur at large scale
  - and which are not observed at smaller scales
- Corollary: In many large networks, even small things can and do cause huge events.
  - In system-theoretic terms, in large systems such as these, even small perturbations on the input to a process can destabilize the system's output.
- In very large systems, even small things can be big

Amplification Examples

- Multiplication of BGP messages across the peering mesh of the Internet
- Tim Griffin’s work showing that even a small amount of inter-connectivity (multihoming, peering) causes the output of a routing mesh to be significantly more complex than its input
- BTW, you see this in nature all the time

And you think the potential for factorial BGP convergence times is bad...

It gets worse...



Aside: Fallacy of Composition

- The logical fallacy of arguing that what is true for the parts is also true for the whole.
  - In particular, there may be amplification occurring in the system

- In the study of economics...
  - This takes the form of assuming that what works for parts of the economy, such as households or businesses also works for the aggregate, or macroeconomy
    - Does this say something about the economics of convergence?

- Example:
  - Both Sodium and Chlorine are dangerous to humans, so any combination of Sodium and Chlorine must be dangerous to humans (NaCl).

Coupling Principle

- The Coupling Principle states that as systems get larger, they often exhibit increased interdependence between components

- Corollary: The more events that simultaneously occur, the larger the likelihood that two or more will interact.

- This phenomenon has also been termed "unforeseen feature interaction" [Doyle and Willinger]

Aside: Fallacy of Division

- The logical fallacy of arguing that what is true for the whole is also true for the parts
  - i.e., Coupling can cause the emergent behavior of a system
- In the study of economics...
  - This takes the form of assuming that what works for the aggregate, or macroeconomy, also works for parts of the economy, such as households or businesses
- Example:
  - NaCl is safe to eat, so Sodium and Chlorine are safe to eat.

Horizontal vs. Vertical Coupling

- Much of the non-linearity observed in large systems is due to coupling
- This coupling has both horizontal and vertical components
- In the context of networking, horizontal couping is exhibited within the same protocol layer, while vertical coupling occurs between layers
- "Layering Considered Harmful"

Coupling Principle Examples

- Canonical examples include Van and Sally's work on the synchronization of various control loops such as
  - Routing update synchronization
  - TCP Slow Start synchronization

- An important result of these observations is that coupling is closely related to synchronization.
  - Injecting randomness is one way to reduce coupling.
  - Another important method for reducing a system's coupling is ensure that local changes have only local effect.

So, what design principles do we have?

- Simplicity Principle:
  - Complexity is the primary mechanism which impedes efficient scaling, and as a result is the primary driver of increases in both capital expenditures (CAPEX) and operational expenditures (OPEX) -- Mike O’Dell (by all accounts)
- But watch out!
  - Everything should be made as simple as possible, but not one bit simpler -- Albert Einstein
- STTA: How does the Simplicity Principle convolve with "policy-based" architecture approaches
  - and more generally, with the Internet Architecture

Example: The Myth of Five Nines...

- NB: Arbitrary Optimization Considered Harmful
  - Why?
- Because optimization introduces complexity and tighter coupling between components and layers
  - Not always: "The best code is no code" [Randy]
  - Virus evolution argument
- The implication here is that trying to squeeze out efficiency past some point only adds complexity, increases non-linearity, and hence leads to less reliable systems (Law of Diminishing Returns)
  - But still, why?

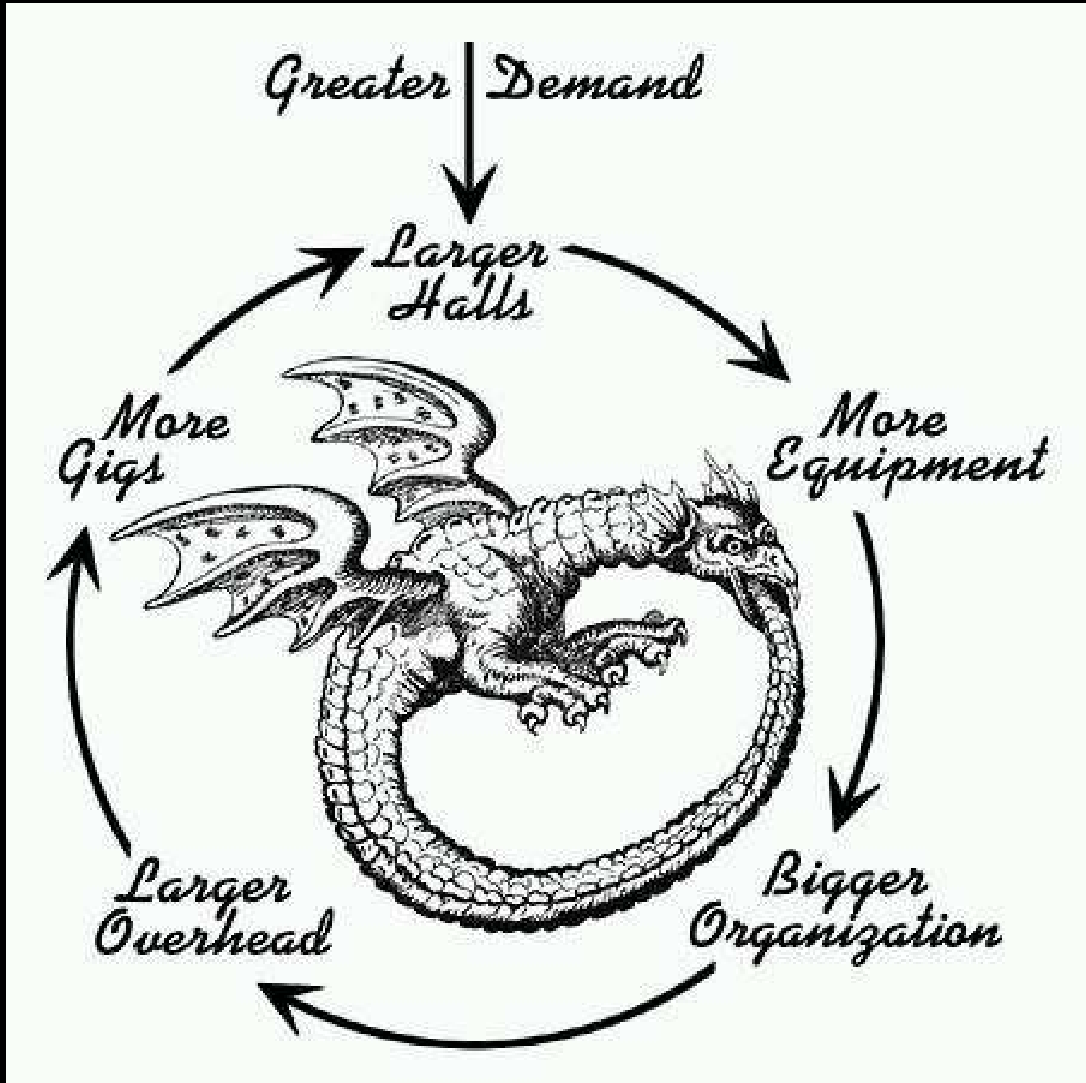
Example: The Myth of Five Nines...

- 80% of unscheduled outages are caused by people or process errors [Scott, many others]
  - That is, there is only a 20% window in which to optimize component reliability
- In order to increase component reliability, we frequently add complexity (which is the root cause of at least 80% of unplanned outages). This effectively narrows the 20% window
- Bottom line: Optimization frequently increases complexity, which increases the likelihood of people and process failure

Complexity/Robustness Spirals

The same phenomenon which makes five nines difficult to reach is also characterized as a "complexity/robustness" spiral [Doyle and Willinger], in which increases in complexity create further and more serious sensitivities, which then require additional robustness, which requires increases in complexity, which in turn requires additional robustness, and so on (hence the spiral).

A "Well Known" C/R Spiral



The creature is called an "Ouroboros" (which means "devouring its tail"). See http://www.dragon.org/chris/ouroboros.html (URL courtesy Sean Doran)

The more things change...

- So all of this is not to say that we can’t do better (we always can). However, it seems that the more things change, the more they stay the same:

- Paul Baran, in his classic 1977 paper, "SOME PERSPECTIVES ON NETWORKS--PAST, PRESENT AND FUTURE", stated that "The tradeoff curves between cost and system reliability suggest that the most reliable systems might be built of relatively unreliable and hence low cost elements, if it is system reliability at the lowest overall system cost that is at issue"

So what is the key takeaway here?

- Complexity isn’t *inherently bad*
  - A system needs some degree of complexity to achieve robustness
  - Contrast with the "simple network" [ISENBERG]
- However, while a system’s robustness is bounded, its complexity is not
  - i.e., the complexity-robustness curve is heavy-tailed
- In other words...

Complexity

Too much or too little, and you're toast

So...what happens if we wind up with...

- A profitable/low margin packet carriage industry

  Hypothesis: e2e principle ==> IP packet carriage is a commodity business
  Profitable/low margin ==> low complexity (Simplicity Principle)
  Economics (+ Science + Technology)

- No access monopolies

  via Competition and/or new technologies
  And policy-based network architectures fail to deliver the margins we hope for?
  Public Policy (+ Science + Technology)

- Large scale attacks against traditional access provider revenue streams

  In the "capture-the-revenue" sense (contrast DOS)
  BTW, recall the IXCs (products of the MFJ)?
  Economics + Public Policy (+ Science + Technology)

Well....

On to Geoff's talk for a look at how all of these things fit together

References

- [Baran] "On Distributed Communications", Paul Baran, Rand Corporation Memorandum RM-3420-PR, http://www.rand.org/publications/RM/RM3420", August, 1964.

- [Doyle and Willinger] "Robustness and the Internet: Theoretical Foundations", John C. Doyle, et al. Draft, March, 2002.

- [Doyle and Willinger] "Robustness and the Internet: Design and evolution", Walter Willinger and John Doyle, 2002.

- [Scott] "Making Smart Investments to Reduce Unplanned Downtime", D. Scott, Tactical Guidelines, TG-07-4033, Gartner Group Research Note, March 1999.