

Decoupling and Computing at Scale – Unlocking the Impossible Triangle

A White Paper Proposal for PeCS 2011

Justin Y. Shi | shi@temple.edu

Introduction

One dichotomy in the networked computing world is that although the networks have experienced rapid growth in scale, reliability and bandwidth, and even drastic latency reductions, robust large scale computing has remained an elusive goal. This non-functional silent problem has become an effective inhibitor for deployment of pervasive computing systems at massive scales.

To date, it is commonly accepted that to get performance, one must sacrifice reliability. To get reliability one must pay with performance. Large scale computer application must accept the increased probability of information losses. Some researchers have even tried to prove these impossibilities are indeed impossible.

This proposal seeks a chance for the communities to take a look at higher level disciplines beyond the natural community boundaries, such as computer architectures, network architectures and application architectures.

Typically, a computer architect begins with the assumption that majority components will not fail. Fault tolerance is the last consideration, if any, in the design process for high performance systems. To date, almost all computer fault tolerance solutions are either cost prohibitive or simply impractical.

Computer application developers follow a similar path. They typically assume that operating systems, database engines and networking layers have taken care of all cases of failures. They design application programming interfaces (API) assume only two possibilities for each action: success or failure.

In comparison, the networking communities have the least assumptions. They must handle failures in the design process. They are all aware that there is such thing called “unknown state”. Fortunately, we can safely rely on the good work of packet switching protocols that afford us a scalable robust data transport plane. Fancier protocols are all based on the robust services underneath.

It should not be hard to see that all communities rely on the networking capabilities. The problems lie in the cross boundary assumptions when the systems are put together. Since the entropy between protocol layers is not yet well understood, many assumptions are simply baseless. They cause “network confusions”. These confusions, we think, hold the keys to our future in robust large scale computing.

The Impossible Triangle



To computer applications and high performance computing architects, performance scalability, non-stop service and lossless information transfer appear to be the three edges of an impossible triangle: a solution to one is likely to be the solution to all.

To network researchers, the packet switching protocols have practically solved these problems for data communication networks almost fifty years ago. Millions of computers are strung together by simple packets. Although we are still facing the future network scalability issues (we are about to run out IP addresses), the addressing scheme may evolve, the two-tier (packet wrapping circuits) basic “store-and-forward” architecture seems destined to stay.

For large scale computing, we believe that the “store-and-forward” protocols have much to offer to our understanding of large scale system’s impossible triangles. It may well hold the secret key, or the network “DNA”, for all our future large scale robust computing efforts.

Proposed Discussions

- What are the key elements in the “store-and-forward” packet protocols that are critical to solving the impossible triangle? Where are the weaknesses?
- Why “store-and-forward” networks are still insufficient for robust large scale computing?
- What are these necessary conditions for robust large scale computing? What about sufficient conditions?
- It seems that whatever solutions we find, they would all be application dependent. This would make it very difficult to form disciplines for in-depth studies and for educational purposes. Are there “basic services” for large scale computing?
- For very large scale deployments, what computer architects must incorporate into their fault tolerance design?
- What application developers must incorporate in their API (and fixing the old) for extreme scale application?
- What the network community must develop to facilitate the tight integration (and to prevent future confusion) between computing, networking and application developments?
- Although the packet switching protocols have practically solved robust data networking problems, security was not part of the design. What can we learn from the network DNA that can be “borrowed” to solve the network security problems?

The philosophical underpinning of the packet switching protocol is something we may call “decoupling” principle. Unlike computer and application architects who are only interested in direct (vertical) high performance channels, decoupling principle is a counter-intuitive holistic (horizontal) method that seeks to disengage critical joints from participating parties in a large system in order to achieve higher level system goals. For the packet switching networks, performance is sacrificed (store-and-forward) to trade for more desirable properties of overall system. Can we deliver DOS (deny of service) attack immune networks using the similar discipline?

In summary, we have proposed higher level philosophical discussions across all disciplines with a focus on applications performance scalability, availability and high fidelity information transfers. The intent is to break the traditional research silos using trustworthy PeSC as the common thread. We believe the NSF/PeSC workshop is an excellent opportunity for effective exchanges amongst deep experienced