Condor: Better Topologies through Declarative Design

A Public Review

Aditya Akella
University of Wisconsin – Madison
akella@cs.wisc.edu

There have been many recent proposals for data center network topologies, and it is difficult for a designer to wade through these options systematically and pick a "good" candidate. Condor is a novel system for data center topology design that attempts to convert this art into science. It replaces traditional manual design with:

- High-level (declarative) expression of building blocks, connectivity, and constraints using a topology description language (TDL).
- Automatic topology synthesis for searching the design space and coming up with candidate topologies.
- Near-automatic testing of generated candidates' key properties against canonical workloads.

This general approach—namely, specifying high level requirements that get compiled down into low level implementations—has been applied with great success in the software world. It has been much harder to realize this vision in hardware. Condor is one of the first systems in this space. Also, the data center arena is an attractive one to attempt this approach, given that the design space is somewhat narrow. It is wonderful that the authors saw this opportunity!

Condor has other exciting features as well. TDL allows precise, yet flexible, expression of key design requirements and constraints. Also, Condor allows operators to express requirements for network expandability (i.e., cost-effective, incremental expansion). Finally, by its very nature, Condor may enable operators to discover novel topologies that are not immediately apparent from manual exploration. E.g., the authors show previously unknown stripings in the fat-tree topology that make it highly fault tolerant.

SIGCOMM reviewers universally liked the problem statement, and the declarative approach the paper takes. The ideas are presented very clearly, with crisp examples backing key claims. The choice of metrics for evaluating generated candidate topologies appears to be grounded in a firm understanding of operational issues in the data center. The evaluation is somewhat limited, but convincing enough in driving home Condor's main strengths.

Yet, the general feeling was that this paper is the first stab at the problem. A variety of issues remain tantalizingly open:

- What are the right metrics to evaluate data center topologies?
- How to generate a suite of workloads that will stress all relevant aspects of a topology? As it stands, test workload generation is ad hoc, and the workloads are somewhat use-case-specific (which is fair, as they are rooted in what data centers run today).
- Is a new description language really necessary? What can TDL not express well? For example, TDL currently cannot express goals that directly address performance metrics. Are there other such examples? What is the right language?
- The operator still has to specify the topology details at a fairly low level, e.g., by specifying the type of topology and connectivity (e.g., that it ought to look like a fat-tree). Is it possible to design an approach that raises the level of abstraction even higher, where the operator specifies high level goals (e.g., cost, reliability, expandability, etc.) and out comes a structure and its instantiation?

Thus, this paper will likely spur a renewal of data center network design research focusing on addressing the above problems, and more!