IEEE ComSoc Distinguished Lecture Tour – Australia, June 8-18, 2014
Ying-Dar Lin, IEEE Fellow
National Chiao Tung University, TAIWAN
July 5, 2014

DLT Planning

This was my first DLT (Distinguished Lecture Tour), piggybacked to IEEE International Conference on Communications (ICC) in Sydney, Australia, 10-14 June 2014. The reason to piggyback this DLT with a conference trip is to save money and time. During ICC, I attended several editorial board meetings, TPC meetings, and technical meetings. Ialso gave the first lecture in University of Sydney, hosted by Prof. Albert Zomaya who is the Editor-in-Chief of IEEE Transactions on Computers. I've known Prof. Zomaya for 3 years since I started to serve on his editorial board. We figured out that we might have a common research interest on an emerging area (software defined networking - SDN) that combines computer society and communication society. Then I flew to Melbourne to give two other lectures at Deakin University, hosted by Prof. Shui Yu, and Swinbourne University of Technology, hosted by Prof. Grenville Armitage. I met Prof. Yu in a conference in Hawaii and he invited me to give a talk in their one-day workshop on Emerging Topics in Computer Science. With Prof. Armitage, Ijust finished guest-editing a successful special issue on Open Source for Networking at IEEE Network. This special issue attracted a record high of 70 submissions and we had to split it into two issues in March 2014 and September 2014.

Though I provided five topics (Research Roadmap Driven by Network Benchmarking Lab, Traffic Forensics, Benchmarking Smartphones, Open Source for Networking, Software Defined Networking) for my hosts to choose from, they all picked the same topic - Software Defined Networking: Why, Where, When, and How. The reason is because SDN is an emerging area and could fundamentally change the networking industry. A 4-min introductory video to the five topics is available at
https://www.youtube.com/watch?v=BuxQ9Yk3OXc&feature=youtu.be.

DLT Itinerary

The itinerary of this DLT, including the meetings in ICC, is as follows. Special thanks should go to the chairs of two local chapters who helped to arrange the local publicity of the lectures:

- New South Wales ComSoc Chapter Chair – Prof. Jinhong Yuan
- Victorian ComSoc Chapter Chair – Dr. Paul G Fitzpatrick

June 8: Depart from Taipei, Arrive at Sydney on June 9
June 10-13: Attend IEEE ICC in Sydney

Technical Program Committee Meeting of IEEE ICC'15
Technical Committee on Computer Communications
Technical Committee on Communications and Information Security
Editorial Board of IEEE Communications Magazine
Editorial Board of IEEE Communications Surveys and Tutorials
Editorial Board of IEEE Communications Letters
Editorial Board of IEEE Wireless Communications

June 11: Talk at 4-5PM in University of Sydney, Building J12
Topic: Software Defined Networking: Why, When, Where, and How
Hosts: Prof. Albert Zomaya, Dr Javid Taheri
Contacts: albert.zomaya@sydney.edu.au, javid.taheri@sydney.edu.au

June 14: Depart from Sydney, Arrive at Melbourne

June 16: Talk at 9-10PM in Deakin Univeristy, School of Information Technology
Topic: Software Defined Networking: Why, When, Where, and How
Host: Prof. Shui Yu
Contact: shui.yu@deakin.edu.au

June 17: Talk at 10:30AM-12:00NN in Swinburne University of Technology, EN 615,Engineering Building
Topic: Software Defined Networking: Why, When, Where, and How
Hosts: Prof. Grenville Armitage, Dr Jason But
Contacts: garmitage@swin.edu.au, jbut@swin.edu.au

June 18: Depart from Melbourne; Arrive at Taipei

Albert Zomaya, his student, and me.

The talk at University of Sydney.


The talk at Deakin University, Melbourne.   After the talk in Deakin Univeristy.

Grenville Armitage, Leith Campbell, and me. After the talk in Swinbourne Univ. of Tech.

Three Lectures

The lecture itself is a tutorial and survey to SDN. I argued why, where, and when for SDN. Then I illustrated how SDN works in four sections: standardization, development, testing, and deployment. These four sections reflect the aspects from standard bodies, vendors, test labs, and operators. Thus, it is probably by far the most comprehensive tutorial and survey on SDN. The lecture has the following title and abstract. It was the final exam week in all universities in Australia. The attendees were mostly faculty members, post-doc researchers, and Ph.D. students. The number of attendees in University of Sydney, Deakin University, and Swinbourne University of Technology was about 25, 50, and 30, respectively.

Title -Software Defined Networking: Why, Where, When, and How

Abstract:

The first wave of cloud computing was to centralize and virtualize servers into the clouds, with a phenomenal result. The emerging second wave, named Software Defined Networking (SDN), is to centralize and virtualize networking, especially its control, into the clouds. SDN deployment started from data centers and now expands to the model of “networking as a service” (NaaS) offered by the operators to enterprise and residential subscribers. By centralizing the control-plane software of routers and switches to the controller, and its applications, and controlling the data-plane of these devices remotely, SDN reduces the capital expenditure (CAPEX) and operational expenditure (OPEX) because the devices become simpler and hence cheaper and number of administrators could be reduced. SDN also enables fast service orchestration because the data plane is highly programmable from the remote control plane at controllers and applications. However, as we detach control plane from where data plane resides, new protocols shall be introduced between control plane and data plane, as the southbound API between controllers and devices and the northbound API between controllers and applications. As we further extend the control plane from controllers to applications such as Service Chaining (SC) and data plane from devices to Network Function Virtualization (NFV), newer mechanisms and APIs need to be added to these APIs. We argue why, when, and where SDN would prevail, and then illustrate how to make it happen. We shall introduce the key technology components, including OpenFlow, SC, NFV, and Network Service Header (NSH) and then review the issues on standardization, development, deployment, and research. At the end, the development and deployment experiences of a campus SDN solution for Wi-Fi/switch control and management are shared.

In-Depth Discussions after Lectures

The lectures triggered many questions from the audience, which also helped me to clarify some issues that I didn’t touch before. I summarize briefly their major questions and my answers below. I also need to follow up with Prof. Zomaya and Prof. Armitage for possible research collaborations on SDN architecture design and SDN test bed design, respectively. The former is on designing and modeling a generic and extended SDN architecture that could provide value-added services beyond connectivity in an efficient way. The latter is on designing the campus-wide test bed to experiment architectural and algorithmic designs.

1. **Why active networking failed and why SDN could succeed?**
   Though both promote the idea of network programmability, active networking tried to put the control of that programmability into every router, i.e., running programs on routers, which is infeasible. In SDN, the control of programmability is the cloud at the controllers, i.e., running programs on controllers to program routers and switches. Cost reduction and new service revenue would be the two driving forces for the success of SDN.
2. **Why redirecting data-plane packets to controllers could lead to performance problems?**
   Most data-plane processing at switches is done in hardware, e.g. table lookup in ASIC, but control-plane processing at controllers and applications is done in software. Redirecting

data-plane packets to controllers would trigger control-plane software processing, which much slows down the forwarding process. Thus, redirection ratio should be reduced.

3. **How can one controller serve a large network?**
   Currently there are about 100K domains on Internet. Some of them would be turned SDN-enabled. Each domain can have one controller or multiple controllers for fault tolerance and load balancing.
4. **How about the Internet backbone?**
   The evolution starts from data centers, then service providers and their subscriber networks they support, i.e., enterprise, residential, and cellular users. It may evolve into handheld and wearable devices. But the entire Internet backbone itself is likely to remain the same, i.e., running BGP routing in a distributed way, because the Internet backbone does not belong to a single domain.
5. **How is the routing information collected in SDN with just OpenFlow?**
   A layer-2 protocol, Link Layer Discovery Protocol (LLDP), enables switches to broadcast themselves and identify their neighbor switches. Then through OpenFlow Hello messages, switches report themselves and connectivity to their controller. The controller takes the collected routing information to construct the topology for path computation and then, through OpenFlow Modify messages, configure flow tables at switches.
6. **Where should traffic classification happen?**
   If only TCP/IP-layer traffic classification is needed, it can be done at switches because the packet headers checked by flow tables are TCP/IP headers. But if application header or even payload is checked in doing traffic classification, it should be redirected to the extended data-plane, i.e.,network function virtualized (NFV) modules. But for the first packet, redirection to the controller for service chaining (SC) is needed to identify where the NFV modules are.
7. **Where should deep packet inspection happen?**
   Since traffic classification might need deep packet inspection (DPI) into application headers and payloads and flow tables do not have entries related to application headers and payloads, DPI should be realized as NFV modules in software. However, this would result in most data-plane packets being redirected to NFV modules. It is possible to move part of DPI back to switches for acceleration. But then switches should maintain data structures beyond flow tables.
8. **Security of SDN vs. security by SDN**
   When talking SDN security, most researchers now talk about securing SDN, especially its centralized controllers which could be the single points of failures. But if we view security as a valuable service that requires resources, the operators could offer SaaS (Security as a Service) to their enterprise, residential, and cellular subscribers. Thus, another stream of research should be on how to offer SaaS on top of NaaS (Networking as a Service) by SDN.