The Abstraction Track

Bringing the SDN Promise beyond Box Limits
Out of the Boxes

- The network does not need to be seen any longer as a composition of individual elements
- User applications interact with the network controller(s)
- The network becomes a single entity
  - Suitable to be programmed
  - Aligned with current IT practices
- We can apply different levels of abstraction
  - Network Stored Program Model
  - Network Operating System
  - Network Abstract APIs
  - Network Abstract Orchestration
- And think of a network design flow
  - And even an IDE
Stored Program. The Network Is *A* Computer

- So we can apply software development techniques and tools
- Software development and operation being multifaceted
  - Different tools for different tasks
  - Some already on their way
- Static and dynamic verification
- Translation, dynamic composition and linking
- Testing and debugging
- Version and configuration control
- Development flows
- And abstraction capabilities
Network OS. SDN in the Widest Sense

- Providing a consistent interface to control, data and management plane
  - A layered model
  - The first take could follow an analogy with existing OSes
- The kernel is realized by control plane mechanisms
- Data plane is associated with the file system
- The management plane is mapped to the system tools
  - Remember the shell
- Specific services to enforce policy and security
  - PAM is a reference
- And the APIs
The Network OS Ecosystem

- The users
  - Network operators
    - Manage the network, create services and locate problems in a more efficient manner
  - Application providers
    - Reduced time to market for new applications, value added services, abstracted view of the network

- The networks
  - Need to address a wide variety of devices and protocols

- The goal
  - To simplify use and management of heterogeneous E2E networks
  - Access, core, datacenter….

- The POSIX reference model
Net-wide, POSIX Style

Application

System Interface - APIs

Filesystem - Data Plane

Kernel - Control Plane

System Tools - Mgmt Plane

OpenFlow

*MPLS (LDP/RSVP)

Policy - Security

L2VPN

IP

LISP

v6

...
Network APIs. Upper Layers of Abstraction

- NaaS beyond itself
  - Current models are still very much box-oriented
    - Virtual view of current elements
- And beyond OpenFlow
  - An excellent practical base
  - As much as processor instruction sets
- A first step: consider the fabric
  - Extend OpenFlow to deal with overlay control
- And start thinking of the equivalents to
  - SQL
  - OO
  - Garbage collectors
  - <YourPreferredITConstruct/>
The Road to a Network IDE

- The natural consequence of applying concepts and tools related to software development
- Supporting a complete design flow
  - High-level definition and manipulation
  - Validation from simulation to actual debugging
  - Beta versions by slicing
  - Phased deployment
  - Integrated with parallel IT development
- Proof of concept
  - OpenFlow-in-a-Box
  - More to come
Abstract Orchestration. SDN Realm Partitioning

- SDN partitioning is inevitable
  - A large network is likely to be divided into multiple SDN realms
  - Each SDN realm with its own controller
- Some reasons
  - Scalability
  - Manageability
  - Privacy
    - Privacy policies applied to tenants or special applicable policies
  - Incremental deployment
- Partitioning is already a common practice
  - FlowVisor-enabled slices
Applying ALTO to SDN Realm Orchestration

- SDN controllers communicate by exporting and importing network information through an ALTO server
- Information exchange is subject to realm-specific policies
- The ALTO server acts as network data orchestrator
  - Control decisions are autonomously taken by controllers
- ALTO as part of an evolved Eastbound (North-East-bound?) API
Making Orchestration Happen

• The ALTO server becomes a “soft” orchestrator
  ▪ No need for a controller hierarchy, mesh, chain, or…
  ▪ Policy driven

• Flexible arrangements
  ▪ Controllers retain autonomy
  ▪ “Multi-homing” is possible
  ▪ And different policies at each attachment link

• Neutrality
  ▪ With respect to positioning in the realm(s)
  ▪ With respect to SDN flavor

• But we need to
  ▪ Decide on extensions to ALTO data models
  ▪ Enhance two-way interactions, session management and timely updates
  ▪ Explore mechanisms for security, discovery, policy declaration, attachment modes…
The Struggle for the Right Abstractions

• We are witnessing a paradigm shift in networking
  ▪ The possibility of interacting with the network as a whole
  ▪ And to reason about that
• Taking the first steps
  ▪ IT is an interesting source of inspiration
  ▪ Its models are limited as well
  ▪ And convergence requires additional effort
• The future of network design and operation lies in building the right abstractions
  ▪ Validation and acceptance are not short processes
  ▪ You can only learn to walk by walking