SDN/DANCES Project Update
Developing Applications with Networking Capabilities via End-to-end SDN (DANCES)

Kathy L. Benninger
Manager of Networking Research
PSC Bettis Briefing

15 September 2015
Agenda

• Review of Software Defined Networking (SDN)
• Overview of PSC DANCES project
• Components and integration
• Lessons learned
The Goals of Software Defined Networking

• Provide advanced networking capabilities that are not readily possible with today’s routing and switching infrastructure
  – Dynamic path provisioning at requested bandwidth
  – Seamless integration with virtual computing environments (e.g., OpenStack’s Neutron)
  – Routing of non-traditional types of network traffic

• Reduce networking costs by creating open source software and enabling “white box” hardware to support SDN

• Give networking researchers, administrators, and users more control over network configuration and operation

• Offer centralized control over complex network topologies
SDN Terminology and Functionality

- SDN/OpenFlow separates network control and data planes
- OpenFlow refers to the communication protocol between control and data forwarding plane elements
  - Provide open, programmatic access to network device flow tables
  - Expand beyond traditional IP routing control to protocol independence (when supported by hardware)
- Data path resides on the switch, on the “fast path”
- Routing decisions are moved to a separate SDN/OpenFlow controller
SDN Terminology and Functionality

• Flow rules are created on the controller and then loaded onto the switch to create the tables for flow processing.

• OpenFlow switch flow tables contain:
  – **Flow entries** against which incoming packet headers are *matched*
  – **Actions** that are then applied to the incoming matching packets
  – Counters
  – Meters

• An incoming flow that doesn’t match a pre-configured rule is sent to the SDN/OpenFlow controller for processing, where it may be dropped or cause a new rule to be added
DANCES Project

• “Developing Applications with Networking Capabilities via End-to-end SDN”

• Project funded through the National Science Foundation’s Campus Cyberinfrastructure program

• Goal is to make the file transfer performance of large data sets more efficient and predictable

• Integrate SDN/OpenFlow network bandwidth scheduling capability with compute job scheduling and distributed file systems.
  – PBS/TORQUE resource schedulers (GridFTP, SLASH2)
  – SLASH2 wide area file system (developed at PSC)
  – XSEDE-wide file system (IBM’s GPFS)
DANCES Project Collaborators

• Funded development sites:
  – Pittsburgh Supercomputing Center
  – National Institute for Computational Science (at University of Tennessee, Knoxville)
  – Penn State

• Deployment and operational testing:
  – Texas Advanced Computing Center (at University of Texas, Austin)
  – National Center for Supercomputing Applications (at University of Illinois, Urbana-Champaign)
  – Extreme Science and Engineering Discovery Environment (XSEDE)
  – Internet2
  – Georgia Institute of Technology
DANCES Network Map
DANCES Functionality

• File transfer is initiated between sites by:
  – User requesting file transfer within Torque script (GridFTP, SLASH2)
  – User accessing the file at a location where the file is not resident (SLASH2, XSEDE wide file system)
  – SLASH2 replication service

• The DANCES control infrastructure will check the following:
  – Does user/project have access to bandwidth scheduling resource?
  – What amount of bandwidth is the user permitted to request?
  – Is that amount of bandwidth available?

• If all three criteria are met, an end-to-end network VLAN path will be provisioned with dedicated bandwidth

• If not, transfer can proceed, but at best-effort throughput
DANCES Functional Diagram

DANCES Participant site

DANCES server and control infrastructure at PSC

FlowSpace Firewall

AL2S

Control Flow

Data Flow

OpenFlow Switch Corsa DP6410

Resource Manager / Scheduler

Control Flow

Data Flow

OpenFlow Switch Corsa DP6410

RYU OpenFlow Controller

Control Flow

OpenFlow Control “Southbound” API

OpenFlow Control “Northbound” API

CONGA bandwidth controller

XSEDE user database XDCDB

Control Flow

Resource Manager / Scheduler

Resource Manager / Scheduler

Data Flow

Wide Area File Server: GridFTP SLASH2

Wide Area File Server: GridFTP SLASH2

Network

© 2015 Pittsburgh Supercomputing Center
DANCES Initiators

DANCES Participant site

DANCES server and control infrastructure at PSC

INTERNET

AL2S

FlowSpace Firewall

OpenFlow Switch Corsa DP6410

OpenFlow Switch Corsa DP6410

OpenFlow Control “Southbound” API

OpenFlow Control “Northbound” API

OpenFlow Controller

CONGA bandwidth controller

XSEDE user database XDCDB

Resource Manager / Scheduler

Wide Area File Server: GridFTP SLASH2

Wide Area File Server: GridFTP SLASH2

RYU OpenFlow Controller

Control Flow

Control Flow

Control Flow

Data Flow

Data Flow

Data Flow
CONGA: Centralized OpenFlow/SDN and Network Governing Authority
Data Transfer

DANCES Participant site

DANCES server and control infrastructure at PSC

FlowSpace Firewall

AL2S

Control Flow

Control Flow

OpenFlow Control "Southbound" API

OpenFlow Control "Northbound" API

CONGA bandwidth controller

XSEDE user database XDCDB

Resource Manager / Scheduler

OpenFlow Switch Corsa DP6410

Wide Area File Server: GridFTP SLASH2

OpenFlow Switch Corsa DP6410

Wide Area File Server: GridFTP SLASH2

RYU OpenFlow Controller

Resource Manager / Scheduler

© 2015 Pittsburgh Supercomputing Center
DANCES Project Status and Lessons Learned

• Install or use existing SDN/OpenFlow infrastructure at each site

• Tested OpenFlow switches:
  – Corsa DP6410
  – HP5920
  – Netronome
  – Juniper EX9204
  – Considered: Arista, Brocade, Cisco, and others...

• Just because a vendor says their switch “supports OpenFlow 1.3” does NOT mean they’ve implemented all the features in the OpenFlow 1.3 spec

• Deploying Corsa DP6410 and HP5920
DANCES Project Status and Lessons Learned

• Run one SDN/OpenFlow controller at PSC – tools to coordinate multiple active controller environments are now becoming available

• Selected the open source RYU controller
  – Python
  – Simple, but includes the functionality DANCES needs
  – Fully supports OpenFlow1.3
  – Broad support

• Differences in switch OpenFlow implementations among hardware vendors require different controller options

• Cross-domain control is challenging
DANCES Project Policy and Operational Issues

• Authentication and authorization to access bandwidth provisioning
• Identification of priority projects and users
• Cooperative agreement between sites on distributed resource control
• Total amount of bandwidth to make available for scheduling
• Bandwidth increments to offer for scheduling
Summary Observations

• The control and flexibility provided by SDN is well suited to certain types of applications
• Carefully evaluate your environment and the problems to be solved
• Many vendors/groups involved in SDN hardware and software development
• SDN/OpenFlow feature support has lagged announced availability dates
• Some vendors of “traditional” networking gear see OpenFlow as a threat
Questions?