Creating a Worldwide Network For The Global Environment for Network Innovations (GENI) and Related Experimental Environments

Joe Mambretti, Jim Chen, Fei Yeh,
International Center for Advanced Internet Research Northwestern University
Chicago, Illinois, USA
j-mambretti@northwestern.edu, jim-chen@northwestern.edu, fyeh@northwestern.edu

Jingguo Ge, Junling You, Tong Li
China Science and Technology Network, Computer Network Information Center, Chinese Academy of Sciences
gejingguo@cstnet.cn, youjunling@cstnet.cn, tongl@cstnet.cn

Cees de Laat, Paola Grosso
University of Amsterdam
delaat@uva.nl, p.grosso@uva.nl

Te-Lung Liu,
National Center for High-Performance Computing,
National Applied Laboratories
Hsinchu City, Taiwan
tlliu@narlabs.org.tw

Mon-Yen Luo
National Kaohsiung University of Applied Sciences Kaohsiung, Taiwan
myluo@kuas.edu.tw

Aki Nakao
University of Tokyo
nakao@nakao-lab.org

Paul Müller
University of Kaiserslautern
pmueller@informatik.uni-kl.de

Ronald van der Pol
SURFnet
Utrecht, The Netherlands
Ronald.vanderPol@SURFnet.nl

Martin Reed
University of Essex
mjreed@essex.ac.uk

Michael Stanton
Brazilian Research and Education Network – RNP
Rio de Janeiro, RJ, Brazil
michael@rnp.br

Chu-Sing Yang
National Cheng-Kung University
Tainan City 701, Taiwan
csyang@ee.ncku.edu.tw

Abstract— **Many important societal activities are global in scope, and as these activities continually expand world-wide, they are increasingly based on a foundation of advanced communication services and underlying innovative network architecture, technology, and core infrastructure. To continue progress in these areas, research activities cannot be limited to campus labs and small local testbeds or even to national testbeds. Researchers must be able to explore concepts at scale – to conduct experiments on world-wide testbeds that approximate the attributes of the real world. Today, it is possible to take advantage of several macro information technology trends, especially virtualization and capabilities for programming technology resources at a highly granulated level, to design, implement and operate network research environments at a global scale. GENI is developing such an environment, as are research communities in a number of other countries. Recently, these communities have not only been investigating techniques for federating these research environments across multiple domains, but they have also been demonstration prototypes of such federations. This chapter provides an overview of key topics and experimental activities related to GENI international networking and to related projects throughout the world.**

1. Introduction

It is well known that the majority of key societal activities are becoming global in scope, and as these activities expand world-wide, they require a sophisticated foundation of advanced communication services, supported by underlying innovative network architecture, technology, and core infrastructure. To continue progress in meeting these and future requirements, network research investigations cannot be limited to campus labs and small local testbeds or even to national testbeds. Researchers must be able to explore innovative concepts at a significant scale -- global scale -- through empirical experimentation. They must conduct experiments on world-wide testbeds that approximate the complex attributes of the real world. Today, it is possible to take advantage of several macro information technology trends, especially virtualization and capabilities for programming technology resources at a highly granulated level, to design, implement and operate network research environments across the world. In the US, the Global Environment for Network Innovations (GENI) is developing such an environment, as are research communities in a number of other countries, described in subsequent sections. [1]

In the last few years, these communities have begun to federate these research environments across multiple domains, in part, to enable wide ranging exploration of innovative concepts at extremely large scales. Also, they have been demonstrating prototypes of such federations at workshops and conferences. Traditionally, network testbeds have been designed and implemented within project frameworks with limited scopes to support fairly narrowly defined research objectives over a short period of time. In contrast, GENI and related testbed environments have been planned to support experimental research across a wide range of topics, as a persistent research resource, within which many topics can be investigated at an extremely large scale – including globally. A notable aspect of these testbeds is that they not only provide a platform for innovative research, but also they incorporate architectural designs, services, and technologies that forecast the basic model of future communications infrastructure. Within these distributed environments, next generation macro trends are emerging, especially those related to the transition from limited static services and infrastructure to unlimited, highly dynamic, deeply programmable, continually evolving innovative environments. Another major transition reflected in the new models is the migration from designing networks that are controlled and managed through proprietary systems closely integrated with proprietary devices to those that are based on open architecture and open systems, for example, using approaches such as Software Defined Networking (SDN) to manage multiple generalized network resources.

Overview of Chapter

This chapter describes the international capabilities of GENI and related network research environments, specifically a) required services for these types of distributed facilities b) basic architectural considerations, c) existing global facilities d) existing international testbed facilities and examples of research experiments being conducted within those environments and f) emerging architecture and design trends for anticipated future services, technologies, facilities, and resource expansions.

The first section provides a brief overview of the required basic services for large scale, highly distributed network science empirical research facilities. A special consideration in this chapter is one that highlights a need for ensuring flexible and programmable multi-domain L2 paths. A common networking architectural model describes seven basic layers. Of these Layer 3 (L3), is the most familiar because essentially, the Internet is based on L3 architecture. However, underlying L3

services are supporting Layer 2 (L2) and Layer 1 (L1 – e.g., lightpaths within optical fiber) services and capabilities that are undergoing a rapid revolution from static to dynamic, programmable resources.

Generally, L2 and L1 paths have been implemented as static resources, implemented without change for long periods. Increasingly, L2 and L1 paths are being implemented as dynamically provisioned paths. Also, providing dynamic L2 and L1 paths across multi-domains requires special considerations because such paths transverse many difference authority, policy, management and control boundaries.

The next section describes basic architectural considerations for large scale research testbeds. Such architectural considerations include those for provisioning dynamic multi-domain L2 and L1 paths as well as hybrid networking paths comprised of services utilizing multiple network layers, e.g., L3, L2, and L1.

The next section highlight existing global facilities, with a focus on the Global Integrated Lambda Facility (GLIF) and its Open Exchanges around the world GLIF Open Lambda Exchanges, or GOLEs) as foundation resources. This distributed facility enables multiple customized production and testbed networks to be created and operated within lightpaths on terrestrial and oceanographic fiber optic cables spanning many thousands of miles. A subsequent related section describes a dynamic networking provisioning API developed by the GLIF community in partnership with a standards organization – the Global Grid Forum. This capability allows customized network to undertake dynamic provision across paths spanning multiple domains world-wide.

The next series of sections describe existing international testbed facilities and examples of research experiments being conducted within those environments. These environments include the international GENI SDN/OpenFlow research testbed, which has been implemented by a consortium of network scientists, the Japanese led international V-Node initiative, a Virtual Research Environment for Large Scale Distributed Systems Research developed by G-Lab in Germany, an international testbed for investigating a variety of topics ranging from WAN protocol transport to Ethernet OAM, and Provider Bridging virtualization, being led by researchers in the Netherlands, a cloud/network testbed being developed in China, a large scale international tesbed for multiple research projects, such as topology management and Virtual Local Area Network (VLAN) transit, a project being led by research institutions in Taiwan, a content routing testbed in the UK, a Brazilian Future Internet testbed, and an international, advanced high performance digital media testbed.

The final sections describe emerging architecture and design trends for anticipated future services, technologies, facilities and resource expansions. Included are discussions of Software Defined Networking Exchanges (SDXs), Software Defined Infrastructure (SDI), which integrates compute resources, storage, instrumentation, sensors, and other resources, and the close integration of network research testbeds and cloud research testbeds.

Required Services

The majority of substantial advances in information technology have been based on innovations that have created a higher layer of abstraction than that which had existed previously. Today, many such major advances are being accomplished as a result of multiple convergent macro trends in information technology that are enabling much higher levels of abstraction and virtualization across all levels of infrastructure. Many are based on Service Oriented Architecture (SOA) and related concepts leading to – Anything-as-a-Service (XaaS), for example, Architecture-as-a-Service (AaaS), Network-as-a-Service (NaaS), Environment-as-a-Service (EaaS), Software-as-a-Service (SaaS), Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), Container-as-a-Service (CaaS), and many more. Recent work by Strijkers et al [2] has even created a model and architecture for an "Internet Factory." Standards organizations are developing open architecture frameworks for these approaches, for example, through the Open Grid Forum's Infrastructure Services On-Demand Provisioning Research Group (ISOD-RG) and the US National Institute of Standards and Technology (NIST), which are developing XaaS open architecture standards [3, 4]. The GENI initiative leverages these trends to create highly flexible, programmable, dynamic, distributed environments. However, the goal of the GENI project is not just to leverage such innovations, but to use them to create an environment that supports services that allow experimental researchers empirically to design, develop, and test concepts that will lead to the next generation of distributed environments.

Although the attributes of next generation distributed environments are still evolving, the current macro trends in design indicate the nature of their eventual characteristics. For example, these trends will allow the creation for multiple highly differentiated networks within the same shared infrastructure, so that networks services can be precisely customized for individual requirements. These capabilities are required for many organizations and organizational partnerships that require private customized and highly specialized network services. However, it is also required by providers of large scale distributed clouds for multiple, perhaps hundreds of individual tenants, each of which requires a private, individually managed and

controlled network. Because no single centralized NOC can support many hundreds of individual networks, new tools are being created to enable self-self network for multi tenant networks. A number of these tools are based on "slicing" architecture, which allows for contiguous integrations of resources, including over international WAN paths, virtual and physical, to be segmented for specialized purposes. As resources are increasingly being abstracted through virtualization, new tools are also being created to discover, integrate, manage and control them, especially through new types of orchestration techniques.

Next generation distributed environments also will allow for much more dynamic network services and infrastructure environments as opposed to today's fairly static implementations, which anticipate basic resources remaining unchanged for long periods of time. They will also allow for a far more granulated control over network resources, including low level resources, such as L2 and L1 paths. As noted, much progress is being made on transitioning L2 and L1 paths from static to dynamic resources and to provide a wide range of enhancements to the capabilities of these paths within compute facilities and data centers, among such facilities and data centers, within metro regions, across nations and across the world.
In addition, they will allow for highly distributed control over network resources, including individual core elements, e.g., ports. Recent progress in virtualization and related in distributed programmable networking, give rise to opportunities for migrating from centralized network control and management to extremely distributed control and management. Management and control functions that previously were the exclusive prerogative of centralized provider NOCs now can be provided to enterprises, applications, processes at the edge of the networks, and individuals.

An especially wide range of new capabilities are being developed for flexible L2 services to meet requirements of such local and wide area deployments. Various technologies being developed include virtualized L2 services such as Virtual Extensible LAN (VXLAN), the IETF's locator/ID separation protocol (LISP), the IETF's Stateless Transport Tunneling (STT), the IETF's virtualization using generic routing encapsulation (NVCRE), the IETF's Network Virtualization Overlays initiative (NVO3), the IETF's Generic Network Virtualization Encapsulation (GENEVE), Multi Protocol Label Switching (MPLS), Virtual Private LAN Service (VPLS – Ethernet type multipoint to multipoint using IP or MPLS, Advanced VPLS (A-VPLS), Hierarchical VPLS (H-VPLS, i.e., using Ethernet bridges at edge and MPLS in the core), Pseudowire (PW – emulation over L3), PW over MPLS (PWoMPLS), PW over Generic Routing Encapsulation, a tunneling protocol (PWoGRE), PW supporting Virtual Forwarding Interfaces (VFI), Overlay Transport Virtualization (OTV), IETF Transparent Interconnection of Lots of Links (TRILL) - link state routing using a routing bridge or TRILL switch, IETF Layer Two Tunneling Protocol (L2TPv3), and others.

Also, many options are being for implementing virtual L2 networks that can be controlled by SDN techniques, for example, L2 VLAN Provider Bridge (802.1Q tunneling or Q-in-Q), Provider Backbone Bridge (PBB - MAC-in-MAC), MEF Access Ethernet Private Line Service (Access-EPL), MEF Ethernet Virtual Private Line (EVPL port-based point-to-point), MEF Ethernet Private LAN (EP-LAN), for port-based multipoint-to-multipoint, MEF Ethernet Private Tree (EP-Tree), MEF Ethernet Private Tree (EP-Tree), port-based rooted-multipoint, MEF Ethernet Virtual Private Tree (EVP-Tree), and MEF Ethernet Virtual Private LAN (EVPLAN).

Even though some of these capabilities are being developed for local (e.g., metro) deployments, they eventually will extend throughout the world. This attribute of extensibility world-wide is the focus of this chapter, which describes how these attributes will characterize international networking services and infrastructure at a global scale.

1. **Global Environment for Network Innovations (GENI) and Related Initiatives**

The GENI initiative, which was established by the National Science Foundation's Computer and Information Science and Engineering (CISE) Directorate, was formulated within the context of the policies of that organization, including those related to international partnerships. [5] Similarly, the European Union's Future Internet Research and Experimentation (FIRE) project has funded a number of major network research testbeds throughout Europe. Within and external to the FIRE program, Europe has established multiple research testbeds, including BonFIRE [6], PHOSPHORUS [7], OFELIA,[8] which is OpenFlow based [9], GEYSERS an optical integrated testbed for ''GEneralized architecture for dYnamic infrastructure SERviceS'' [10], FEDERICA [11], and the G-Lab testbeds [12]. G-Lab, which is presented in one of the chapters of this book, and one of the experimental areas supported in presented in a section in this chapter, has a wide ranging agenda including research projects on a functional composition concept for a dynamic composition of functional blocks on network and service level, topology management, and investigations of federation concepts to interconnect with international Future Internet Testbeds. The European Future Internet project, FED4FiRE, is developing federation techniques for networking testbeds. [13] GENI and FIRE have been federated. In China, the Chinese Academy of Science has established network research testbeds, such as the Sea-Cloud Innovation Environment, through the China Science and Technology Network (CSTnet) to support future network research. In Taiwan, the National Center for High-Performance Computing in partnership with the Taiwan Advanced Research and Education Network (TWAREN) has established multiple network

research testbeds related to Future Internet initiatives. In Japan, National Institute of Information and Communications Technology (NICT) has supported the New Generation Network initiative for many years, particularly through projects based on the JGN-X testbed. In addition, individual institutions have established a number of related projects such as the National Institute of Advanced Industrial Science and Technology's G-Lambda project [14] and the University of Tokyo's V-Node project. In Korea, the K-GENI project has been established as a persistent multi-topic research testbed [15] In South America, the primary area of focus for these projects has been multiple Brazilian Future Internet research and development projects, including FIBRE [16].

In Canada, the Strategic Network for Smart Applications on Virtual Infrastructure (SAVI) was established as a partnership of Canadian industry, academia, research and education networks, and high performance computing centers to investigate key elements of future application platforms. [17] SAVI, which has designed and currently operates a national distributed application platform testbed for creating and delivering Future Internet applications, is described in a chapter of this book. The primary research goal of the SAVI Network is to address the design of future applications platform built on a flexible, versatile and evolvable infrastructure that can readily deploy, maintain, and retire the large-scale, possibly short-lived, distributed applications that will be typical in the future applications marketplace. GENI has been federated with SAVI, which supports multi-domain interoperability. In additional, GENI has been federated with several cloud testbeds, such as those supported by the NSFCloud program, i.e., Chameleon and Cloudlab, which are described in a later section of this chapter.

Basic Architectural Considerations

The GENI environment is a distributed instrument, which can be used by researchers to discover and claim resources ("slivers"), to integrate those resources within private research environments "slices," conduct experiments using that slice measure and analyze results, and, importantly, reproduce specific results. Note that the GENI architecture is discussed in a chapter in this book. The ability to replicate experiments is a key research requirement. A primary component of the GENI environment consists of the SDN/OpenFlow architecture, protocols and technologies. [9]. The OpenFlow model is part of an instantiation of a number of broad architectural concepts.

Currently deployed digital communication services and technologies comprise the most significant advances in the history of communications. At the same time however, they are based on architectural approaches that are beginning to demonstrate major limitations that restrict future progress. For example, in today's networks, control and management functions are implemented with an assumption that the communications environment will be fairly static - that they will remain unchanged for long periods of time. Network control planes are have only limited scope based on minimal state information, such as neighbor connections, reachability, and access policy. This approach cannot meet rapidly changing requirements and demands for on-going dynamic service and technology enhancements as well as for quick adjustments to network resources tin response to changing conditions. Consequently, a new architectural approach is being developed to provide increased capabilities for programmable networking, especially the set of techniques termed Software Defined Networking (SDN). This model, which separates the control plane from data plane, provides for programmability and a higher level of network control abstraction and enables a more comprehensive overview of network state information, which can be used to dynamically control networks services and resources.

This comprehensive overview is made possible by an ongoing dialogue between network devices, including individual components, and controllers. Instead of having state information confined to individual devices within the network, this information is gathered by logically centralized controllers. The network devices send the controllers state information and the controllers use that information to make decisions on dynamically matching demand requirements with resources, on solving problems such as sudden congestion, and allocating resources in anticipation of demand because of network behaviors. These decisions are signaled back to the network devices for implementation, for example by programming flow tables in network devices. Using this approach, network devices can be considered undifferentiated component resources, and the specialized capabilities can be provided by the control plane. Because these controllers are logically centralized they have a global view of the network and consequently, they can provide for much better traffic optimization than is possible using traditional distributed approaches.

Currently, the most common implementations of these concepts are based on OpenFlow, which enables controllers to have access to a set of network primitives. The actual capabilities for programmable networking are provided by control frameworks. The general GENI architecture and has several major control frameworks (ORBITS, ProtoGENI, PlanetLab, the Open Resource Control Architecture – ORCA, and the GENI Experiment Engine - GEE) as well as an SDN implementation

and also a general Aggregate Manager API developed to integrate these frameworks and its mesoscale facility implementation are described in other parts of this book.

Just as the GENI AM integrates and coordinates the control frameworks within its environment, it is possible to consider a type of international aggregate manager that functions more universally, including as part of a multi-domain federation comprised of many international network research testbeds. There are many techniques available that can be used to support such interdomain interoperability, including using the GENI control frameworks, such as ProtoGENI/InstaGENI, ORCA/ExoGENI, and combinations.

This chapter explores the implications of such federations, particularly the special considerations required for international network connections among multi-domain federations of major experimental network environments. A primary set of considerations with regard to such international federations relates to the interoperability of, and in some cases, the direct integration of, the individual control frameworks for the individual environments being connected. An important, closely related, set of considerations concerns multi-domain international connections as opposed to those dealing with international single domain networks. The majority of the core architectures currently being used for these environments, such as SDN/OpenFlow, are single domain and not multi-domain oriented. Because highly distributed multi-domain models are much more complex than single domain, the majority of the topics discussed in this chapter relate to multiple domain architecture and implementations.

To accomplish such federations, a number of elements are required, including those providing resources that can be made available to experimenters, implementing a means for advertising those resources, discovering them, claiming them, e.g., through reservations, discarding them after their use, and managing the components of the federated environment, including addressing problems. In addition, other mechanisms are required, e.g., for designing and implementing state machines, interacting with those state machines, signaling messages, interpreting those messages, sharing topologies, determining path finding, path stitching, deploying and using resource ontology and schemas, gatekeeper interfaces, federation gateways, SDN exchange points (SDXs), and more. In addition, given the objective of interdomain resource sharing, processes must be implemented with appropriate policy to drive security mechanisms including those for authentication and authorization. Overall, mechanisms must be established for APIs, secure signaling, resource identification, advertising and discovery, trust relationship management, trust root services, federation policy enforcement, certification, monitoring and analytics, and related functions.

Although all major network research environments in various countries undertake to design and implement these basic capabilities, all approaches are somewhat different. However, because of the recent progress in virtualization of networks and in control plane capabilities, opportunities exist to develop such federations despite such the differences among architectural approaches.

Creating a Common International Network Language and Network Programming Languages

In order to facilitate projects spanning different services, domains, and infrastructures, methods must be created that allow exchange of information about available resources and state information so that such resources can be requested and allocated for use by network services and applications. One such infrastructure information model is the Network Description Language (NDL) [18] pioneered by the University of Amsterdam that forms the basis of the Network Markup Language Workgroup in the Open Grid Forum (OGF). The NDL provides a method to describe computer networks in a meaningful way. The NDL ontology for computer networks uses the Resource Description Framework (RDF) [19]. With this ontology one can create a simple, clear, understandable description of a network. The Network Description Language (NDL) helps to reduce the complexity issues in computer networks. The goal of NDL is to allow not only network processes but also applications to have a better understanding of the network so they can more easily adapt it to their needs. NDL has been extended to include descriptions of computing and storage resources: the Infrastructure and Network Description Language (iNDL) [20]

Some research groups are interested in extending these types of languages to resources beyond networks. For example, the ORCA-BEN [21] project developed the NDL–OWL model, which uses the Web Ontology Language (OWL) instead of RDF, extends NDL to include cloud computing, in particular, software and virtual machine, substrate measurement capabilities and service procedures and protocols. This ontology models networks topology, layers, utilities and technologies (PC, Ethernet, DTN, fiber switch) based on NDL. In comparison, INDL uses the latest developments in the OGF NML-WG. Standards

organizations are continuing to evolve such languages so that there can be meaningful information exchanges among infrastructure related processes. Such standards are particularly important for network inter-domain provisioning.

Programmable networking requires a data modeling language. Network management protocols generally have related data modeling languages. For example, the first Internet network management tool, Simple Network Management Protocol (SNMP), utilizes Structure of Management Information (SMI), incorporating Abstract Syntax Notation One (ASN.1). When the IETF was developing the NETCONF protocol, it was observed that a data modeling language was needed to define data models manipulated by that protocol. In response, the IETF developed YANG ("Yet Another Next Generation"), a modular data modeling language for the NETCONF network configuration protocol.

The following sections describe several approaches being undertaken by research communities to create techniques for inter-domain federations. The sections immediately following this one describe some of the international foundation resources, especially those based on lightpaths implemented within world-wide optical fiber, that are being used for that research.

Existing International Facilities

1. *Global Lambda Integrated Facility (GLIF)*

One major global facility that is being used to support multiple distributed international environments for network research is the Global Lambda Integrated Facility (GLIF) [22] The GLIF is a world-wide distributed facility designed, implemented and operated by an international consortium, within which participants can create many types of customized services and networks, including those that are required to support international network research environments. Unlike most communication exchange facilities today, which interconnect only at Layer 3, the GLIF was designed to enable networks to exchange traffic at all layers, including Layer 1. GLIF is based on a foundation of owned and/or leased optical fiber paths and lightpaths within optical fiber, including trans-oceanic fiber. Lightpaths are created and managed through technologies and services based on Dense Wavelength-Division Multiplexing (DWDM), which supports multiple parallel high performance, high capacity channels.

The GLIF environment is highly complementary to the GENI environment because it was designed for network programmability. (Ref: Figure 1) GLIF domains are interconnected by the GLIF exchange facilities – Open Lambda Exchanges (GOLES), which have implemented different types of control frameworks. Current GLIF exchange points are: (AMPATH (Miami), CERNLight (Geneva), CzechLight (Prague), Hong Kong Open Exchange - HKOEP (Hong Kong), KRLight (Daejoen), MAN LAN (New York), MoscowLight (Moscow), NetherLight (Amsterdam), NGIX-East (Washington DC), NorthernLight (Copenhagen), Pacific Wave (Los Angeles), PacificWave (Seattle), PacificWave (Sunnyvale), SingLight (Singapore), SouthernLight (São Paulo), StarLight (Chicago), T-LEX (Tokyo), TaiwanLight (Taipei), and UKLight (London). A related international facility is the Global Ring Network For Advanced Applications Development (GLORIAD), which is directly interconnected with the GLIF and supports international network testbed research and other application level projects. [23] A consortium including the US, Russia, China, Korea, Canada, The Netherlands, India, Egypt, Singapore, and the Nordic countries supports GLORIAD.
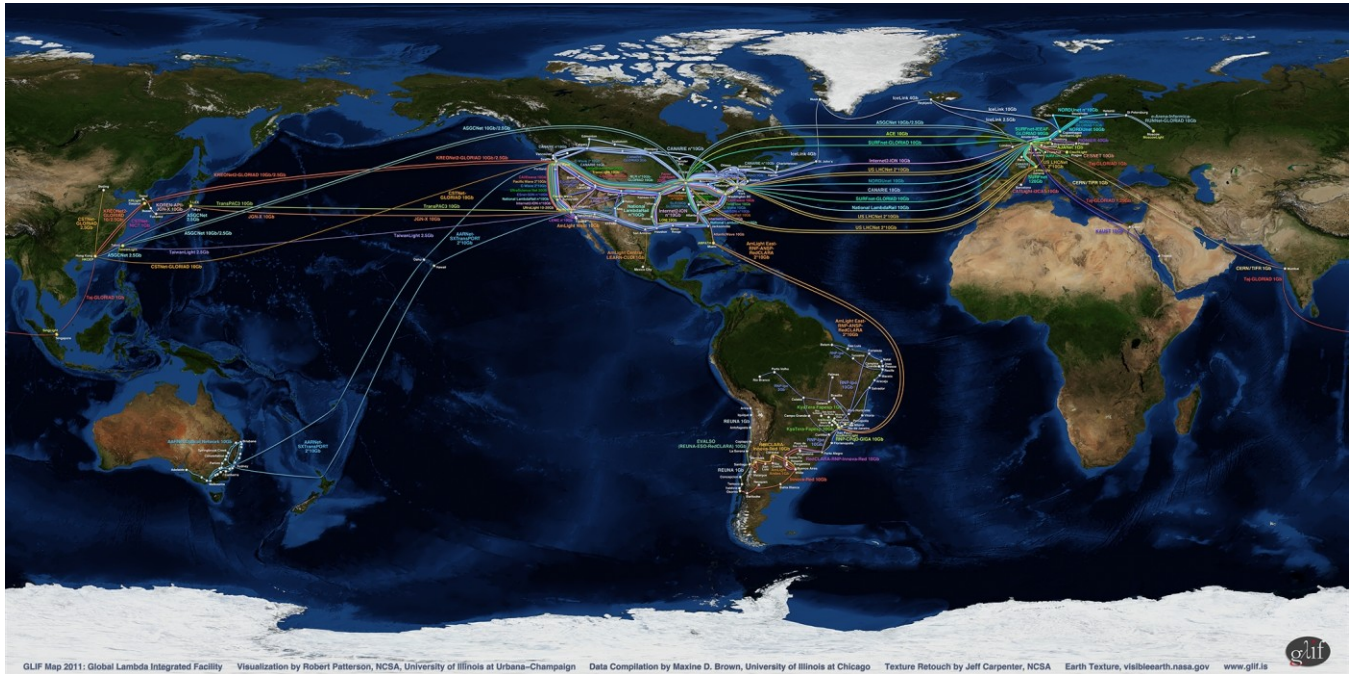
Fig 1: Global Lambda Integrated Facility (GLIF)

Network Service Interface – NSI Connection Service

Because the GLIF community is comprised of multi-domains and because uses multiple different control frameworks for dynamic provisioning, an initiative was established by that community with a standards development organization -- the Open Grid Forum (OGF) -- to create architectural standards for a generic network service interface (Network Service Interface – NSI) as an API to the multiple control frameworks within the GLIF environment, including an NSI Connection Service (currently v 2.0). [24] When the GLIF exchange facilities around the world were implemented (GLIF Open Lambda Exchanges or GOLEs), they were established with multiple different control framework for resource management and control, for example, DRAC (Dynamic Resource Allocation Controller), Autobahn, Argia, OSCARS (On-Demand Secure Circuits and Advance Reservation System), G-Lambda, and many others. In other words, each open exchange point had a different control framework for reserving and establishing links through the exchange point. The SI enables these paths to interconnect. This service was designed specifically to assist the creation of multi-domain connections across international networks and through these exchange points using a common API that would allow provisioning across multiple networks operated by many different national research and education network organizations. The NSI specifies signaling, state processes, messages, protocols, and other environmental components. A process or application at the network edge can discover and claim network resources, at this point primarily paths, within an environment comprised of heterogeneous multi-domains. [24]

After several years during which the GLIF NSI participants demonstrated persistent international testbed capabilities, especially through the AutoGOLE series of demonstrations (Ref: Figure 2), NSI implementations are being placed into production for a number of national R&E networks. This figure illustrates available VLANs implemented among multiple GLIF GOLEs that are available for use by communities participating in the AutoGOLE initiative.

Although this service is not yet being used extensively today to support network research environments, it is worth mentioning here because plans are being developed to do so, and because the NSI connection standard already has developed many of the mechanisms required to support interconnectivity among multi-domain network research environments. Also, several GOLES have started to incorporate SDN/OpenFlow capabilities to support L2 based traffic.
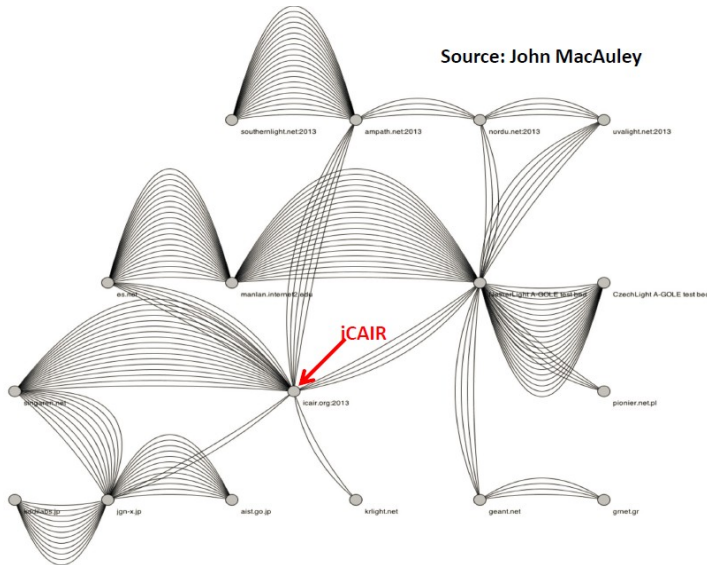
Figure 2: AutoGOLE SC13 Demonstration Schematic Showing Addressable, Programmable Paths Among Global GLIF Open Lambda Exchanges (GOLEs)

The NSI initiative established a working group that has examined many architectural issues within the Network Services Framework (NSF) specified in the OGF GWD-R-P Network Service Framework v2.0. This framework defines an outline for a set of protocols. The NSF expects that that resources and capabilities can be advertised externally through defined Network Services, and it defines a unified model for how various processes should interact with such services, for example, creating connections (Connection Service), sharing topologies (Topology Service) and performing additional services required by a federation of software agents (Discovery Service). NSI allows for implementing network paths across multiple network domains operated by disparate network providers, enabling federations. The NSI architecture specifies Service Termination Point (STP) objects, which are used by connection requests to determine connection attributes. The STP is a means to abstract resource functionality at the point where NSI services terminate from actual underlying physical resources and configurations, such as nodes or circuits. Such abstraction made possible by STP allows for use of functional options multi-domain transport termination without forcing users to deal with the complexity of the hysical infrastructure and configurations at the termination points. An STP is the designation of a specific topological location that functions as the ingress/egress point of a network. The STP has a definition as a single Service Type. An STP can be a single termination point or a group of STPs. Such a set is termed a Service Domain. STPs within Service Domains can be completely interconnected. Service Domains also can be interconnected. Adjacent and connectable STPs (that is, one or more pairs of STPs with matching attributes/capabilities) managed by separate networks can interconnect at a Service Demarcation Point (SDP). [24] (Ref: Figure 3)
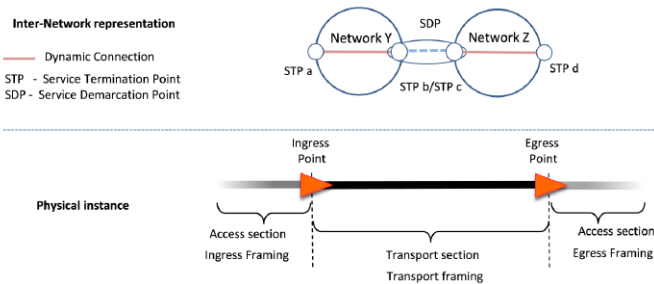

Figure 3. STPs Interconnecting at a SDP.

1.1. The International GENI (iGENI) and the International Advanced Network Research Facility

The International GENI (iGENI) initiative was established to create a federated international network research testbed facility. A number of iGENI participants and other international network research partners collaborated in the designed and

implementation an International Advanced Network Facility based on SDN/OpenFlow as a research platform to provide network research communities with world-wide experimental resources, including addressable transport paths, that can be used to investigate many different types of topics. [25] The platform also provide options for closely integrating programmable networks with programmable clouds [26] The figure below depicts a topology that was showcased at SC11, SC12, SC13, SC14, SC15, at multiple GENI Engineering Conferences, at other technical and research workshops and at other events (Ref: Figure 4).

Figure 4: International GENI (iGENI)/International OpenFlow Testbed Topology

This international federated testbed was designed to enable researchers to discover, claim, integrate, use, and discard a variety of diverse resources, including core network resources, as slices across a shared infrastructure fabric. Each site has a collection of resources that are interconnected with a mix of dynamic and static L2 VLANs. Tools and methods for undertaking these tasks through orchestration frameworks are being developed. Such orchestration is one of the components of a hierarchical architectural stack, with edge process signaling, which could be application signaling as well as system process signaling, at the top of the stack.

This international federated testbed is being used to explore various techniques for designing and implementing orchestration processes are used to discover and claim segments of network resources, including full topologies, and options for configuring those segments, to dynamically provision paths and endpoints, and to specify the specific attributes of the services that they create. Some projects are focused on developing northbound access to the processes that control and manage the actual resources, discovery, claiming, accessing state information, configuring, provisioning, etc. Other projects are investigating southbound interfaces that provides network resource request fulfillment and state information on resources.

Several projects are investigating eastbound and westbound interfaces (E-W), which are key resources for establishing federated interoperable paths across multiple domains, for example, supporting message exchanges among controllers, including those on reachability across multiple domains, controller state monitoring and fault condition responses, and multi-domain flow coordination. E-W interoperability mechanisms for federation are discussed in more detail in a later section of this chapter. Currently, the most widely deployed E-W federation protocol at L3 is the Boarder Gateway protocol (BGP), an Autonomous System (AS) peering path vector protocol use to supporting TCP/IP network exchanges.

Other projects are investigating tools for monitoring, measuring and conducting analytics, e.g., to validate and to verify the stream attributes, e.g., the performance of new types of communication services and to provide stream behavior real time information to traffic engineering and optimization processes.

Resources made available through this platform can be made available to multiple groups of researchers at the same time so that they can conduct experiments with interfering with each other. The figures below illustrate this type of resource segmentation on this platform used for SC11, SC12, and SC13. (Ref: Figure 5, Figure 6, Figure 7). Each color represents a different project undertaken by a difference research group. The figures give an indication of the distribution of resources across the various projects. Subsequent sections describe research projects that have been or that are being conducted on this platform.
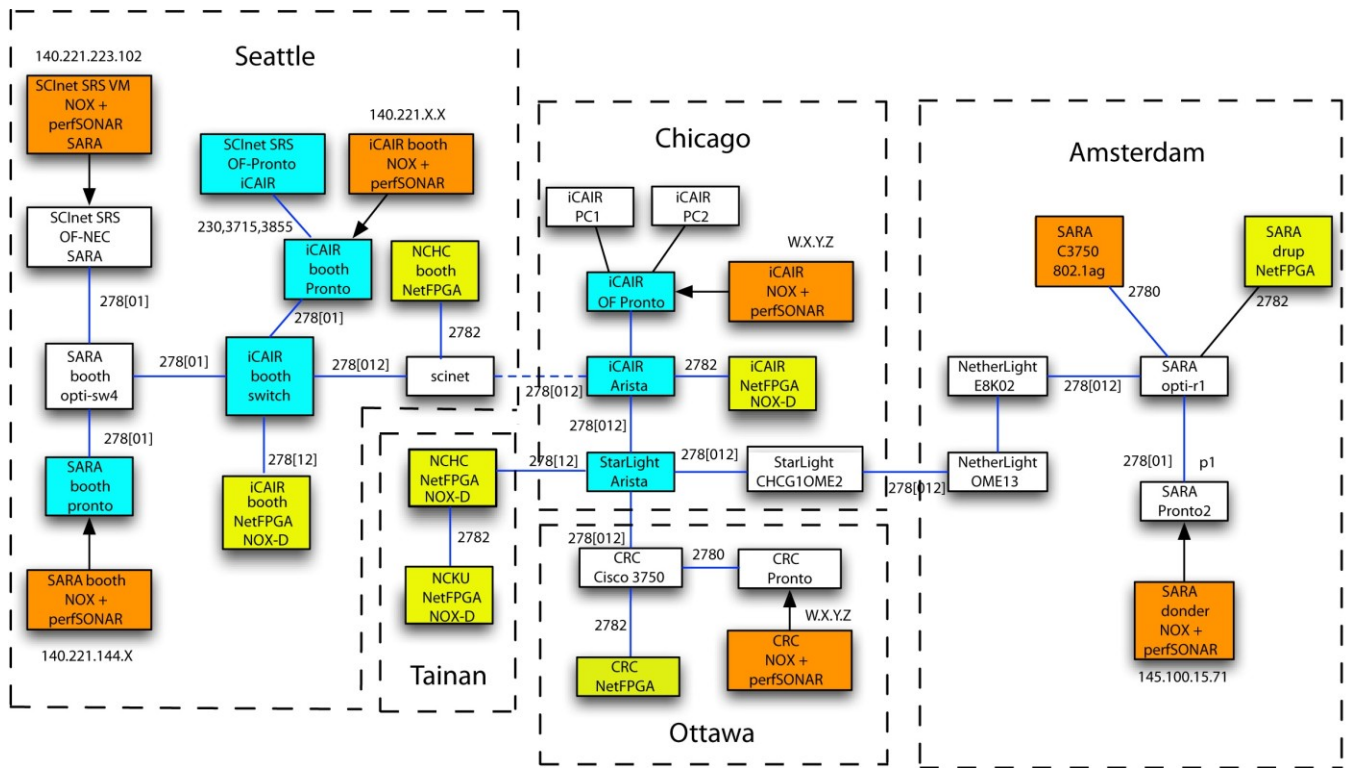

Figure 5: International Advanced Network Research Facility Segmented By Project at SC11
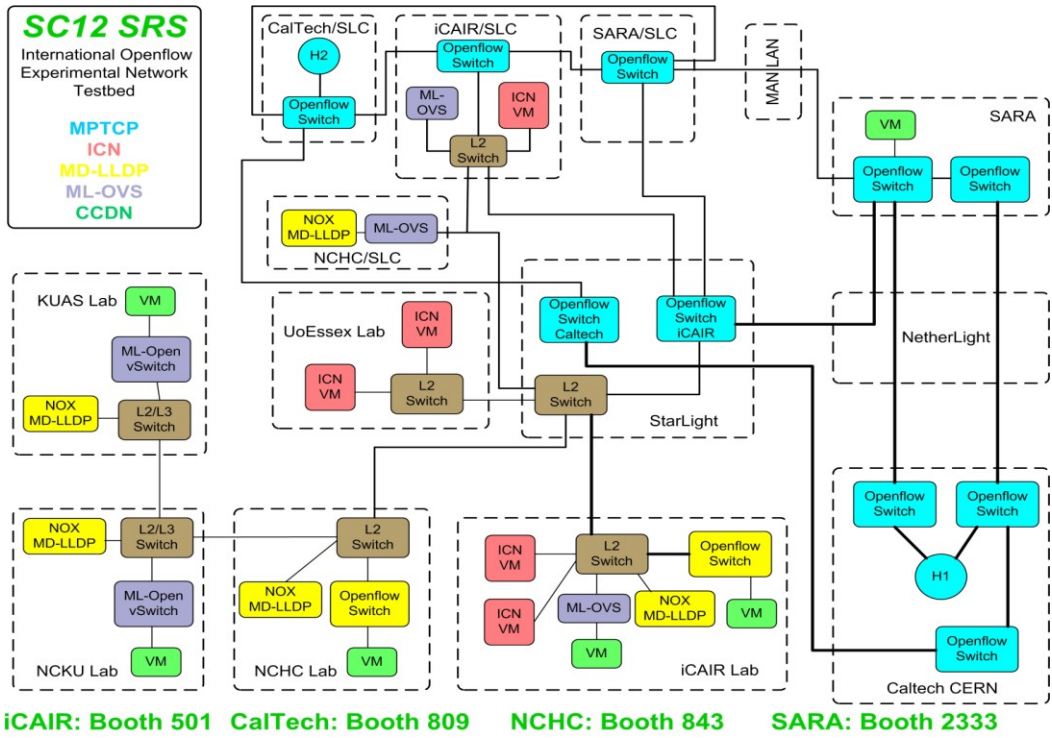
Figure 6: International Advanced Network Research Facility Segmented By Project at SC12
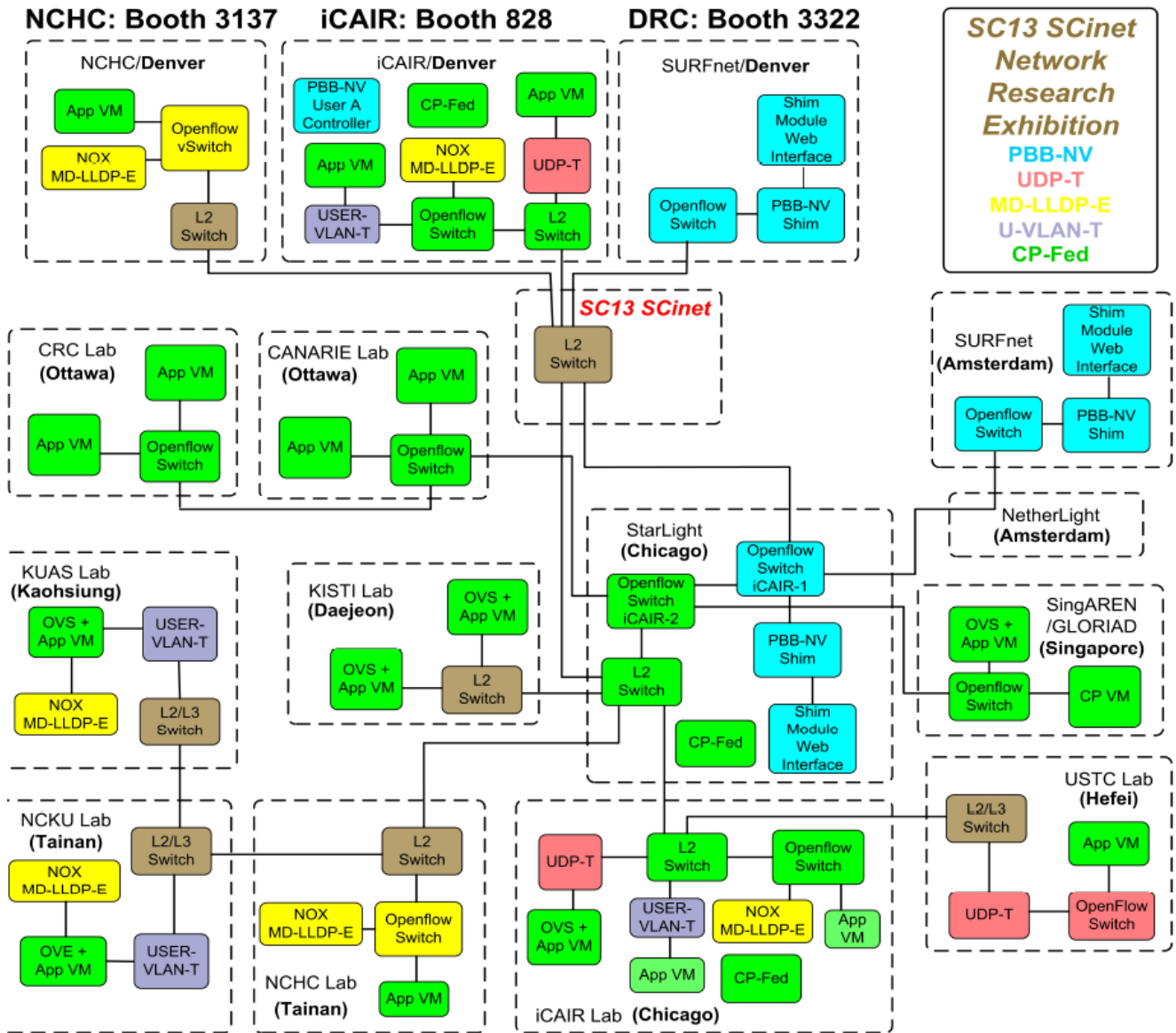
Figure 7: International Advanced Network Research Facility Segmented By Project at SC13 Research Activities and Experiments Conducted Among Current International Environments

Slice Around the World Initiative

Different network research communities are developing various techniques for virtualizing distributed environments and networks, for integration, for developing control frameworks, designing network middleware and for integrating resources. Consequently, federation among these capabilities has become a major research topic.

The 'Slice Around the World" initiative was established to both provide a large scale research platform and to demonstrate the powerful potential of designing and implementing world-wide environments consisting of multiple federated international computational and storage clouds closely integrated with highly programmable networks. As a basic capability, this initiative created a distributed, integrated OpenFlow environment interconnected through a customized international network. All sites have servers capable of supporting addressable VMs. Among the sites, there is a blend of static and dynamic resources. Various aspects of the design for this initiative were considered, including three primary components: a) showcasing one or more application capabilities, for example, some aspect of federated cloud based digital media transcoding and streaming as opposed to merely showing bit-flow graphs b) demonstrating the capabilities of programmable networks using OpenFlow, and c) designing a network architecture based on an international foundation infrastructure. Each of these components is

further described in a subsequent section of this description. Also, participants in this initiative are developing a number of innovative architectural and basic technology concepts. Goals of the project include providing striking visuals, reflecting the potentials of a truly global environment, closely integrating programmable networking and programmable compute clouds, show capabilities not possible to accomplish with the general Internet or standard R&E network, highlighting the power of programmable networks, especially customization at the network edge, and showcasing a potential for resolving real current issues.

For example, for several Slice Around the World (SATW) demonstrations, Finite Difference Time Domain (FDTD) distributed simulation visualization capabilities have been demonstrated using the SATW distributed environment. FDTD is one of most commonly used computational electrodynamics modeling techniques for many research and industry simulations, such as LSI design electro verification. Under current HPC workflow techniques, researchers submit jobs, retrieve results, visualize those results and then resubmit the job with modifications, additional information, data, etc. Today this is a tedious, manual slow process, in part, because of the limitations of today's networks. These SATW demonstrations showed how by using dynamically programmable networks closely integrated with computational and storage clouds, it is possible to provide capabilities that can be used to create interactive simulation/visualization instruments to significantly improve this traditional process. Interactive real-time simulation/visualization instruments included: a) distributed back-end MPI rendering clusters and storage, b) a web front end to setup control parameters for rendering and display the result, c) customized web server to pipe rendering results to users efficiently, and d) a program to check the rendering result and submit jobs if the results were not produced. Web interfaces were used to dynamically identify the sites around the world, where the simulation images located, to convert the request and to send the request to the appropriate host over the private international network, and interactively visualize the simulation over the private network specifically designed for the demonstrations.

Another series of SATW demonstrations used the TransCloud international distributed testbed incorporating programmability for a range of resource infrastructure. [27] The TransCloud is a world-scale high-performance cloud testbed, incorporating a lightweight slice based federation architecture and a slice-based federation interface, with high-performance dedicated intersite networking enabling high-bandwidth data transport between physically distributed sites, the use of experimental transport protocols and guaranteed QoS among distributed clouds, lightweight, robust isolation between components. TransCloud supports both network researchers and researchers developing new types of efficiently managed virtualized computing aggregates, including researchers creating extensions of cloud control environments such as Eucalyptus, Tashi, OpenStack, and VICCI.

The initiative was established by a number of network research centers and labs that are participating in multiple next generation networking activities, including those developing large scale distributed experimental network research environments. Participants have included ANSP, the Applied Research Center for Computer Network at Skolkovo, Chinese Academy of Sciences/CSTNET, the Communications Research Center, Ottawa, CPqD, Duke University, ETRI (Electronics and Telecommunications Research Institute), G-Lab, TU Kaiserslautern, Hewlett Packard Research Labs, the International Center for Advanced Internet Research at Northwestern University, KISTI, KUAS/NCKU, NCHC, NICT, NICTA, Princeton University, the Renaissance Computing Institute, RNP, SURFsara, the University of Amsterdam, the University of Essex, the University of Tokyo, and the University of Utah.

International V-Node

Another major Slice Around the World initiative was the International V-Node project, organized by the University of Tokyo. The V-Node (Virtual Node) initiative was established to enable deeply programmable networks, especially for experimenting with arbitrary protocols, by creating extremely virtualized infrastructure, including supporting multi-domain implementations. The V-Node architecture provides for federated multi-doimain control and data planes and for federation among multiple virtualization platforms. Architectural components include a Gatekeeper (GK) and a Federation Gateway (FGW), which provide for translating API messages, ensuring common data for APIs, and packet formatting. The architecture also includes a Slice Exchange Point (SEP), which supports bridge commands, control frameworks, and policies. One international V-Node SATW demonstration implemented a V-Node at the University of Utah next to a ProtoGENI node, demonstrating the integration capabilities of the V-Node federated functions. Another demonstration, based on the V-Node architecture demonstrated an innovative packet caching technique that provided for hashing of data packets to enable optimized data transport and routing, that is, converting data to hash values and determining responses based on the analytics of those values.

ToMaTo a Virtual Research Environment for Large Scale Distributed Systems Research

Multi-domain federations require topology management tools. The topology management tool (ToMaTo) [28, 29] has been developed as part of the German-Lab project [30], which has been funded by the German Ministry of Education and Research (BMBF) to provide a virtualized research environment for networking experiments. This tool has been implemented on a large scale international testbed. Currently, the ToMaTo testbed is run by a consortium and academic institutions can join the testbed without cost if they contribute resources. Therefore, ToMaTo continuously grows and already spans across multiple continents.

ToMaTo is a topology-oriented networking testbed designed for high resource efficiency, i.e., high parallelism where possible but high realism where needed. Topologies consist of devices (produce and consume networking data) and connectors (manipulate and forward data). Devices contain a set of networking interfaces that can be connected to connectors. Figure 1 shows a simple topology consisting of five devices (one central server and four clients) and three connectors (two switches and one Internet connector). (Ref: Figure 8) To increase both flexibility and resource efficiency, ToMaTo offers different types of devices and connectors. Users can choose between hardware virtualization, which provides an environment nearly identical to a real computer but has a high resource usage, and container virtualization that uses fewer resources but does not suit all needs.
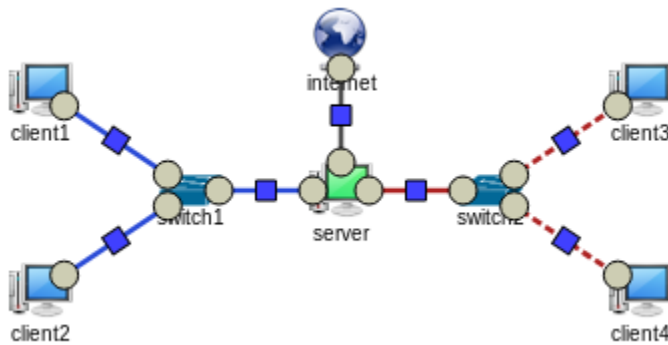

Figure 8: Example Topology

The default connector type is a VPN connector with a selectable forwarding policy (hub, switch or router). Public services, cloud resources or even other testbeds can be combined with ToMaTo topologies by using external network connectors. To help users with running their networking experiments, ToMaTo offers an easy-to-use, web-based front-end with an intuitive editor. Users can control their devices directly from their browser or using a VNC client of their choice. Advanced tools like link emulation and packet capturing are included in ToMaTo and can be used to run experiments. (Ref: Figure 8)
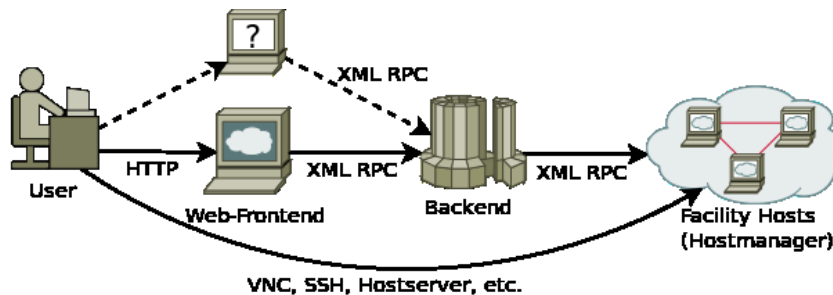

Figure 9: ToMaTo's Three Tiers Architecture

The ToMaTo software consists of three tiers as shown in Figure 9. The hosts provide virtualization technology and a complete toolset needed for advanced features like link emulation, packet capturing, etc. The back-end component contains all the control logic of the ToMaTo software and remotely controls the hosts. Different front-ends use the XML-RPC interface provided by the backend component. The most important front-end is the web- based user front-end that allows users to

edit and manage their topologies from their browser using modern web technologies. Other front-end software includes a command line client that allows easy scripting and an adapter for the other federation frameworks.
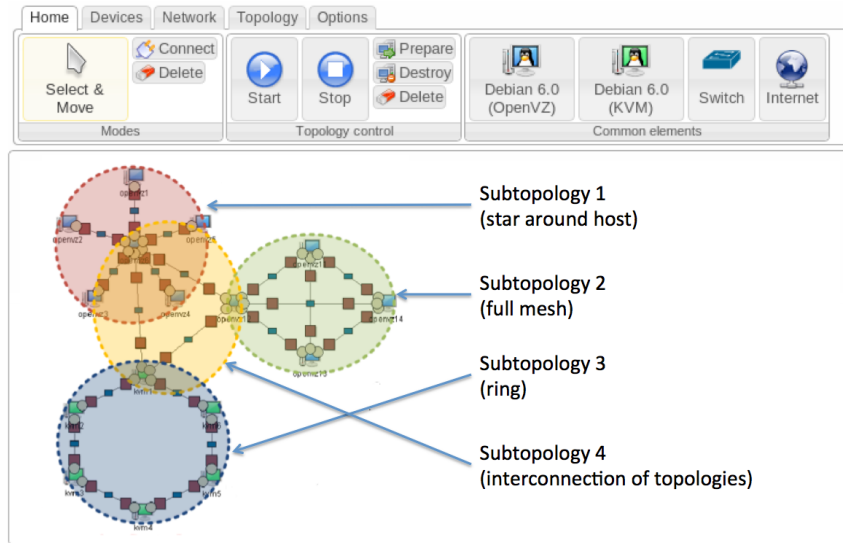

Figure 10: ToMaTo Topology Editor

One of the key features of ToMaTo is its graphical user interface that allows even inexperienced users to create complex network topologies by drag and drop. ToMaTo features an easy-to-use editor (Figure 10) to create, manage, and control networking topologies. With this editor, users can configure their topology components and control them in the same intuitive interface. The editor also gives users direct access to their virtual machines using web-based VNC technology.

Monitoring OpenFlow Slices With Ethernet OAM

During SC11, an international project led by SARA, and including several partners (SURFnet, iCAIR and CRC) showed how the IEEE 802.1ag Ethernet OAM protocol could be used in an OpenFlow controller to monitor and troubleshoot an OpenFlow slice. The slice spanned two continents and included a couple of strategically placed OpenFlow switches in Amsterdam (NetherLight), Chicago (StarLight), Ottawa and the venue in Seattle. (Ref: Figure 5). The SDN NOX OpenFlow controller was used to manage the OpenFlow switches. An open source implementation of the 802.1ag protocol was added as a module to the NOX SDN controller. Using this implementation, the controller could send Ethernet OAM hello frames between the switches. When there was a "fiber cut, which was staged as part of the demonstration, these hello frames were lost and the OpenFlow switch at the other side of the link detected this fault and reported the link as being down. A monitoring website periodically retrieved the link status of all links from the controller and showed the status of the network in real time on a web page.

Multipath TCP (MPTCP)

During SC12, an international project led by SARA, and including several partners (Caltech, iCAIR, and SURFnet) demonstrated a technique for supporting large eScience data transfers over a multipath OpenFlow network [31, 32]. Data sets in eScience are increasing exponentially in size. To transfer these huge data sets, it is important to make efficient use of all available network capacity. Usually, this means using multiple paths when they are available. In this demonstration, a prototype of such a multipath network was implemented. Several emerging network technologies were integrated to achieve the goal of efficient high end-to-end throughput. Multipath TCP was used by the end hosts to distribute the traffic across multiple paths and OpenFlow was used within the network to support the wide area traffic engineering. Extensive monitoring was part of the demonstration. A website showed the actual topology (including link outages), the paths provisioned through the network and traffic statistics on all links and the end-to-end aggregate throughput.

Multipathing is usually undertaken based on flows by calculating a hash (including, e.g., Ethernet addresses, IP addresses, and TCP/UDP port numbers) of the packets. Flows with the same source and destination follow the same path. When the traffic has many different flows the traffic will be evenly balanced over the different paths. But all the paths need to have the

same bandwidth. Another disadvantage is that in large data eScience applications there are typically only a few flows and hashing will not spread the load evenly across the paths in those cases. Multipath TCP (MPTCP) tries to solve these limitations. MPTCP can handle paths of different bandwidth because there is a congestion control mechanism across the subflows. This congestion control mechanism also makes sure that traffic on a congested link is moved to a link with less congestion. Therefore, it adapts the load balancing according to the load of other traffic on the links. In this demonstration, MPTCP was used in combination with an international OpenFlow based multipath network. Data was transferred from CERN in Switzerland through the StarLight International/National Communications Exchange Facility in Chicago to the SC12 venue in Salt Lake City. OpenFlow switches were placed at CERN, NetherLight, StarLight and the SC12 booths of Caltech, iCAIR and the Dutch Research Consortium. (Ref: Figure 6) An OpenFlow application connected via a controller to the OpenFlow switches and automatically discovered the topology via LLDP. The application calculated multiple paths between the servers and the forwarding entries for the flows were pushed to the OpenFlow switches. The demonstration showed the success of using MPTCP for large scale data transport.

Provider Backbone Bridging Based Network Virtualization

During SC13 in Denver, SURFnet and iCAIR demonstrated how Provider Backbone Bridging (PBB) could be used as a network virtualization technology in OpenFlow networks (see figure 5). An important use of Software Defined Networking (SDN) is network virtualization or slicing. This technique allows multiple users to be supported by the same physical infrastructure, with each having their own virtual network or slice. FlowVisor is one of the options to achieve this result. FlowVisor is a software module that is implemented between an OpenFlow switch and OpenFlow controllers and it gives each controller a part (slice) of the flowspace. The disadvantage of this approach is that controllers do not have access to the full OpenFlow tuple space and therefore the capabilities are less than if direct access to a physical OpenFlow switch was provided. Also, at this time, FlowVisor supports OpenFlow 1.0 only and not later versions of the protocol, such as OF v.1.3.

In this demonstration, Provider Backbone Bridging (PBB) as defined in IEEE 802.1ah was used, as encapsulation technology in the OpenFlow data plane. In this way user traffic was separated and identified by the I-SID in the encapsulation header. The data part was the user's original Ethernet frame and users could create OpenFlow rules that matched on any of fields that OpenFlow 1.3 supports, except for the PBB~I-SID because this element is used to map packets to users. This approach is a simple virtualization method that gives users access to virtual OpenFlow switches that have the same OpenFlow capabilities as physical OpenFlow switches would have. During SC13 in Denver, this PBB based network virtualization was shown on the OpenFlow enabled link between NetherLight in Amsterdam and the StarLight facility in Chicago. Pica8 3920 OpenFlow switches were used at both sides. These switches supported OpenFlow 1.3 and PBB encapsulation and decapsulation.

The Sea Cloud Innovation Environment

The Sea-Cloud Innovation Environment (SCIE), a national wide testbed initiative supported by the "Strategic Priority Research Program - New Information and Communication Technology"(SPRP-NICT) of the Chinese Academy of Sciences, is focused on building an open, general-purpose, federated and large-scale shared experimental facility to foster the emergence of new ICT. Recently, plans have been discussed to extend this environment to international sites using federation techniques over international network facilities such as GLIF and GLORIAD. To support the principle proposed for adaptive service-oriented experimentation, SCIE developed a wide-area testbed with hardware resources including servers, cloud services, and storage resources located geographically at five cities including Beijing, Shanghai, Shenyang, Wuxi, Hefei in China (Ref: Fig. 11).
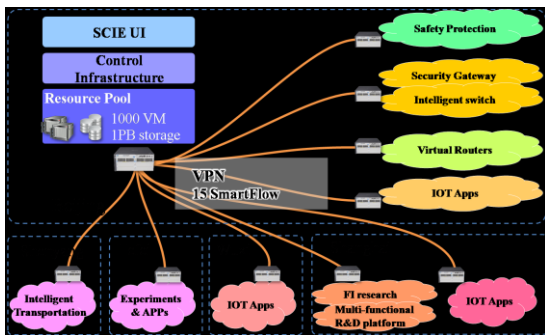
Figure 11. The Deployment of SCIE

The SCIE is connected by Smart-Flow devices which support open flow 1.2 protocol. The GRE-tunnel enabled Smart-Flow device can encapsulate layer-2 protocol inside virtual point-to-point links over an Internet. Moreover, we add QoS feature to Smart-Flow device to offer link efficiency mechanisms that work in conjunction with queueing and traffic shaping to improve efficiency and predictability of virtual links. To decrease the complexity of experiment device deployment, SCIE Rack is designed and prototyped. Like the ExoGENI and InstaGENI, the SCIE Rack provides integrated control and measurement software, network, computation, and storage resources in a single rack.

Beyond the network and hardware layer, the SCIE offers distributed resource control, experiment measurement and an experiment service system, which was developed by the CSTNET research and development team. This system gives researchers a graphic user interface that can be used to design and use virtual network topologies consisting of virtual devices and links over the SCIE testbed. A distributed experiment traffic analysis measurement tool offers researchers to subscribe traffic statistics and virtual link performance data by BPF-like syntax. Moreover, to help researchers control their experiments on SCIE, Java and Python develop libraries are provided for researchers who can run their experiment codes on SCIE experiment playground or their own devices. The SCIE architecture is shown as Figure 12.
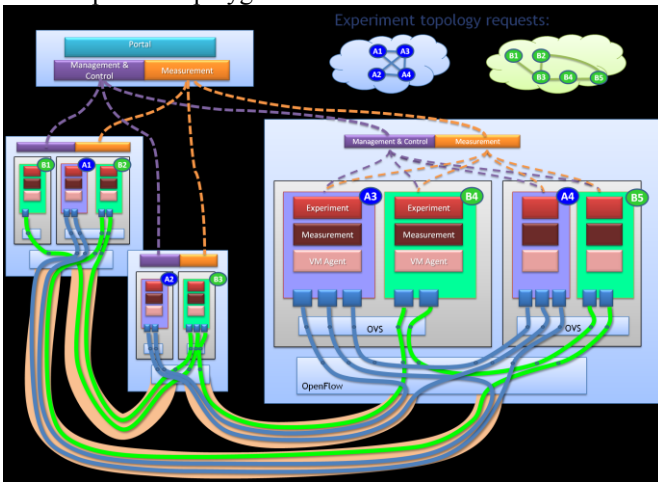

Figure 12. SCIE Architecture

The key technologies of SCIE can be summarized as follows:

Resource virtualization: the SCIE uses both slice-based and time-based virtualization to handle sliceable and un-sliceable resources, respectively. Sliced resources can be requested and dynamically created for researchers, where the request description and visual rendering for experimentation can be achieved in SCIE (see Fig. 4).
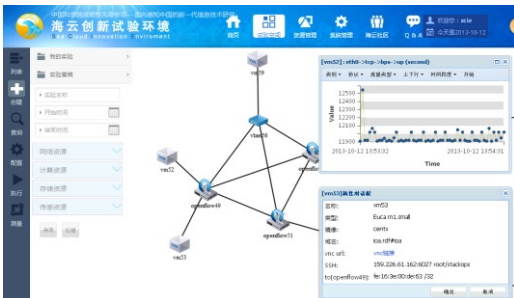

Fig. 4.

Request description and visual rendering for an experimentation using SCIE.

High-level experiment work flow description programming language: The SCIE offers two types of libraries for high-level programming languages: Java and Python to describe the experiment work flow, measurement data subscription and

experimentation visualization. With an experiment control program, the experimenters can schedule tasks running on slivers and subscribe the measurement data they need, and also, they can use all the characteristic of these languages.

Holographic measurement: To decrease the impact on experimentation, the SCIE mainly leverages passive measurement, and still provides active measurement to design the holographic measurement system. In particular, the environment adopts several tools, e.g., sFlow and Spirent TestCenter SPT-3U testers, and uses both real-time flow handling and offline flow processing engine to provide data collection and analysis for experimentation.

International Multi-Domain Automatic Network Topology Discovery (MDANTD)

In almost all current OpenFlow deployments, there is a single controller managing all switches and handling network topology for routing decisions. However, for large-scale OpenFlow environment that interconnects several network domains, a single-controller may cause performance downgrade problem and policy/management issues. The National Center for High-Performance Computing is developing mechanisms to address such scenarios, by enabling each domain to deploy its own controller and to exchange topology and traffic information with other controllers. [33] Today, OpenFlow lacks an east-west interface standard (i.e. interface between controllers vs north south communications with resources within a single domain), and there are no signs that any standard body plans to work on this issue in near-term. Without such interface, when a network has problems or if the flow policy encounters some error, managers of each controller can only know what happen in their control domain, i.e. they cannot get any error information about other domains. In such situation, it is hard to investigate the problem accurately and the network manager will take more time to troubleshoot the network or flow policies.

In order to resolve the problem, NCHC has designed and implemented multi-domain automatic network topology discovery for large-scale OpenFlow/SDN environment by simply modifying Link-Layer Discovery Protocol (LLDP). LLDP, defined as IEEE 802.1AB, is a vendor-neutral link layer protocol for discovering neighbor devices. Most of Openflow controllers adopt LLDP for automatic topology discovery by sending LLDP packets to switches in its domain periodically. Upon receiving the packet, OpenFlow switch will forward to its neighbor switches and then send back to controller. Controller will analysis traveling path of each LLDP packet and conclude the network topology of its domain.
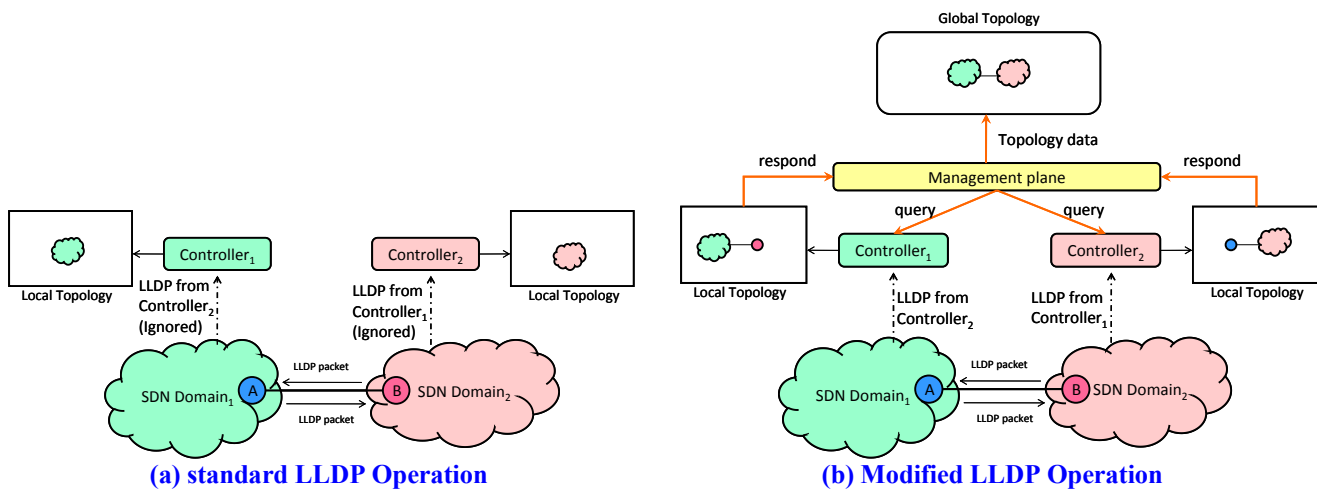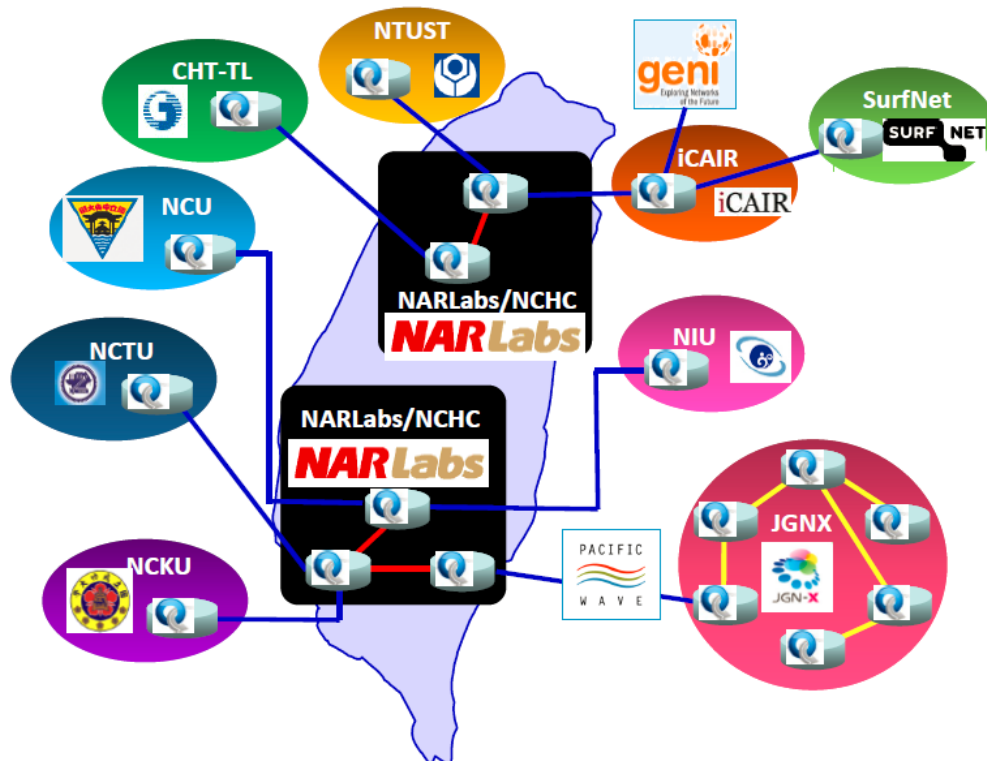

(a) standard LLDP Operation   (b) Modified LLDP Operation

Figure 13: Modifying LLDP for Inter-Domain Automatic Topology Discovery

In Multi-Domain OpenFlow networks, when a LLDP packet travels to the controller of another domain, it will be ignored and dropped. Hence, the topology information cannot be exchanged. For example in Figure 13(a), there are two SDN Domains where $Controller_1$ manages $Domain_1$ while $Controller_2$ manages $Domain_2$. Switch A in $Domain_1$ has an inter-domain link with Switch B in $Domain_2$. LLDP packets from $Domain_1$ will travel from A to B and then will be sent to $Controller_2$. $Controller_2$ will ignore and drop the packet because it comes from another domain. As a result, $Controller_1$ only knows $Controller_1$'s topology and $Controller_2$ only knows $Controller_2$'s topology separately.
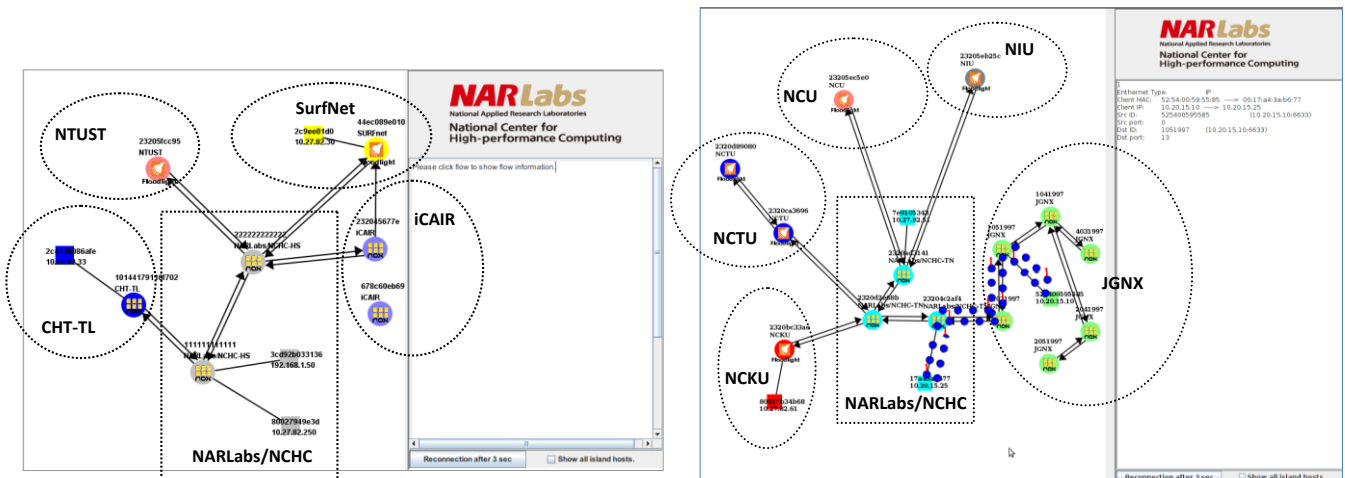
This new approach modifies LLDP operation as illustrated in Figure 13(b). When $Controller_2$ receives the LLDP packet from $Domain_1$, it will pick it up and analyze the traveling path. $Controller_2$ hence learns that there exists an inter-domain link from $Domain_1$'s A to $Domain_2$'s B. Finally, $Controller_2$ knows $Domain_2$'s topology with an inter-domain link to switch A. Similarly, $Controller_1$ knows $Domain_1$'s topology with an inter-domain link to switch B. A separate management plane has been designed

that contacts each controller to obtain its local topology. Global topology is then understood by inter-connecting each domain's local topology and shown on a GUI console. This algorithm has been implemented in both NOX and FloodLight SDN controllers so that the management console can display multi-domain networks with a mix of NOX and FloodLight controllers.

In the latest demonstration of this technique at SC13, 7 domestic institutes in Taiwan (NARLabs/NCHC, NCKU, NCU, NTUST, NIU, NCTU, CHT-TL), StarLight/iCair, SURFnet, and JGN-X participated in a large scale inmternational implementation. As depicted in Figure 14(a), the network was divided into a north part and a south part. Figure 14(b) and Figure 14(c) shows the UI display of north part and south part separately. Each Domain is displayed in distinct color for easily identification. Circle nodes represent OpenFlow switches while square nodes represent user's end-node. NOX or FloodLight icons are placed in circle nodes so that managers can recognize which controller is adopted in this domain. In addition, a process intercepts PACKET_IN event to record the flow status and display end-to-end flow in the same user interface. In Figure 14(c), there is a ping issued from NARLabs/NCHC to JGN-X. The end-to-end path is shown in UI and the flow detail is given in the right-side panel.

(a) Demonstration Topology

Figure 14: Multi-Domain Automatic Network Topology Discovery in SC13 Demonstration

In conclusion, this project developed a multi-domain automatic network topology discovery mechanism, and proved that it runs on cross-controller platform by implementing over NOX and FloodLight. It is notable that the mechanism can be incorporated into almost any SDN controller, such as the increasingly popular OpenDayLight, and many others. For large-scale OpenFlow/SDN environment that inter-connects several network domains, managers can easily observe network status and troubleshoot flow status with our management console. In the forthcoming year, this work will be extended to design inter-domain flow provisioning functions. These activities are key functions to implement OpenFlow/SDN network exchange centers in the near future.

Future Internet Virtualization Environment and VLAN Transit

With support from a project award by the National Science Council of Taiwan government, researchers at NCKU and KUAS have been designing and maintaining a network testbed that interconnects several heterogeneous datacenters across the public Internet. The current implementation and deployment has supported several research projects and applications, including cloud federation [34], international testbed interconnections [31], and network/security research experiments over production networks [35, 36]. With the system support, each sliver service could be allocated a portion of network resources that may be hosted either on the hosts within a datacenter or on a collection of physical hosts across different datacenters. As a result, a network system is required to support automated management (creation, deletion, and migration) of virtual networks, large numbers of attached physical and virtual devices, and isolated independent subnetworks for multi-tenancy. The novel contribution of this project is toward the design of a system to transform networks among a collection of independent autonomous datacenters into a manageable, flexible, multitenant transport fabric.

The current deployment depicted in Figure 1 illustrates the challenges of this project. Datacenters in this testbed are primarily interconnected with shared, public layer-3 networks (i.e., the Taiwan Academic Network (TANet) and Taiwan Advanced Research and Education Network (TWAREN)). With support from iCAIR at Northwestern University and the National Center for High-Performance Computing (NCHC), these testbeds have also been extended to connect with the iGENI project in the United States. To isolate the traffic, some of datacenters (e.g., NCKU, KUAS, NCU in this Figure 15) applied some Virtual Private LAN Service (VPLS) paths to interlink with the other systems. However, VPLS only offers a partial solution to the aforementioned problems. The first reason for this shortcoming is that the use of 12-bit VLAN IDs limits the solution's scalability. As a result, in this environment, only a few VLAN IDs are allowed to be used to interconnect all sites, which cause another serious scalability and performance problems. The system suffers from issues caused by flooding mechanisms (e.g., ARP broadcast), the huge MAC table size, and the use of Spanning Tree Protocol in a huge layer-2 network. Third, the VLAN, or VPLS, does not have an easy way to manage the configuration of paths across several network domains.

The architecture of this system can be logically viewed in three layers of a software stack, which also provides the hooks to other middleware and serves as a control plane to orchestrate all operations of virtual networks. First, they abstract the data plane of both the physical and virtual switches and synergize them coherently to enable the dynamic configuration and management of virtual networks [37]. They design a software module that resides in the kernel or hypervisor of each host in a datacenter. They also extend a POX Python based SDN controller with some added function to orchestrate the software modules to create a distributed data plane layer. The second layer is designed to deal with the interconnection issues among multiple datacenters. The major building blocks of this layer include a gateway switch that is designed based on the OpenFlow protocol. Each datacenter should at least have one gateway system located on the edge of its network to relay traffic across sites [38]. The operation of the gateway system is based on a novel mechanism termed virtualtransit [39]. Each pair of datacenters may set up one or more forwarding paths such as VPLS or MPLS. The basic idea of the virtualtransit is to abstract these paths and dynamically translated the VLAN tag to relay the traffic to the selected datacenter. A distributed control plane layer provides inter-datacenter mechanisms to create and manage virtual private networks spread across public Internet. The third layer is a logically central control framework containing the global network topology and the policy that determines the behavior of the whole system [39]. The control framework performs route computation on the virtualtransit and explicitly installs flow information to the switches along the chosen route based on OpenFlow protocol. Consolidating all flow decisions to the controller framework, a central service management system can be implemented to enforce a policy throughout the whole system. The system can also be extended further to add new functions and features by simply updating the controller. For instances, it provides interface to allow integration with other control frameworks such as ORCA for user authentication, resource registration and discovery.
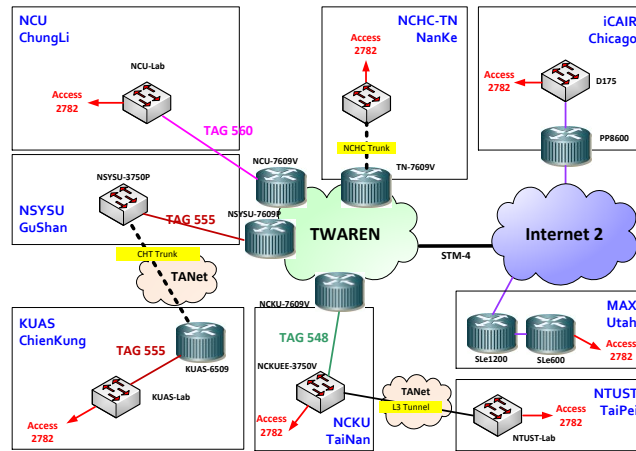
Figure 15: Network Topology of Deployment

Interdomain ExoGENI

As noted in other chapters of this book, ExoGENI is an advanced computer and network resources virtualization project that paves the way for a new wave of virtualized applications that utilize elastic infrastructures. Basically, ExoGENI allows the IaaS paradigm to be extended to include networks integrated with the common compute and data processing cloud resources. The infrastructure information model used by the ExoGeni community for information exchange is based on the NDL, described in an earlier section of this chapter. The use of Semantic-Web technology in this approach facilitates the creation of models that can be easily connected, stacked and extended by other models. Also, as noted earlier, NDL has been extended to include descriptions of other types of resources, such as computing and storage, through the Infrastructure and Network Description Language (iNLDL). The extensibility and applicability of iNDL has been clearly demonstrated by its use as a basis for modeling efforts in three different infrastructures: the CineGrid (digital cinema production processes) infrastructure, [40], the NOVI [41] federated platforms, and the GEYSERS architecture (Generalized Architecture for Dynamic Infrastructure Services) [10].

Currently, a project is being designed that will enable a distributed ExoGENI environment, with sites in the US and the Netherlands to investigate new capabilities for using SDN techniques for the international provisioning of extremely high capacity individual data streams, including streams with 40 Gbps-100Gbps capacities across thousands of miles, across nations and between continents. General networking provides support for supporting aggregations of 10s of millions of small data flows. This project is directed at creating new capabilities for supporting extremely large scale individual flows, using underlying resources described by NDL and orchestrated by SDN methods.

Content Routing

New large scale networking testbeds provide many opportunities for fundamentally changing traditional concepts. For example, the traditional Internet is based on physical addressing, although the Internet is primarily used for information gathering. This observation is one of the motivations behind increasing interest in alternative Internet architectures based upon content based forwarding to replace, or augment, traditional IP routing [42]. Content-based forwarding is often termed content centric networking (CCN) or information centric networking (ICN), and there is not, currently, one model that can claim to fully describe this evolving area. However, one of the central aims of most of the content-based forwarding approaches is to provide an architecture that concentrates on *what* the data is, rather than *where* it is placed. This new approach allows optimization of the network architecture to suit the delivery of the data rather than an optimization based upon end-node attachment, as is inherent in the existing IP routing strategy. Indeed it has been shown that using a content-based forwarding approach allows forms of traffic engineering that are challenging to implement in contemporary IP or IP/MPLS networks [43]. To further explore the area of ICN, the EU FP7 project PURSUIT (leading on from the EU project PSIRP) has implemented a *clean-slate* publish-subscribe ICN architecture [44]. The PURSUIT architecture has a semi-centralized mediation layer and makes use of source routing based on Bloom filters. Using this architecture, the project has demonstrated a scalable ICN solution capable of forwarding at up to 10 Gb/s using standard software based computer platforms; it promises the possibility of forwarding in custom hardware at much higher rates and with lower complexity than IP or IP/MPLS.

The PURSUIT project has created an experimental testbed that is managed at the University of Essex (UK) with partners connecting in globally (EU, US, Japan, and China). As this architecture is exploring forwarding using non-IP based mechanisms it is important that connectivity can be provided at layer-2. One of the experiments carried out as part of the project used multiple end-points and forwarding nodes hosted at Starlight (Chicago) and at the University of Essex (UK). The experiment demonstrated *information* resilience, enabled through the inherent anycast nature of the PURSUIT ICN architecture. Using this approach, a subscriber obtains content from any available publisher. If there are multiple publishers then there is a natural resilience available as the content is fetched through the content identifier, not by the end-point. Specifically the experiment between Starlight and the University of Essex demonstrated resilience for a video transmission. It first showed traditional network resilience: if a network link becomes unavailable the architecture can route around the failed link. As a further step, all links to the video sender were cut, the architecture detected the failure and selected an alternative sender as a source of the video. A key point of this *information* resilience is that it is enabled in the network functions and requires no support from the application itself. ICN projects such as PURSUIT, which are proposing *clean-slate* Internet architectures, require flexible testbed networks that allow layer-2 connectivity. It is clear that, looking towards the future, flexible testbeds enabled through systems such as GENI will be a vital first step towards deployment of future architectures such as ICN.

Brazilian Future Internet Experimental Environment

FIBRE (Future Internet testbeds/experimentation between BRazil and Europe) [45] was one of five projects that were approved in response to the 2010 Brazil-EU Coordinated Call in ICT, jointly funded by CNPq (the Brazilian Council for Scientific and Technological Development) and by the 7th Framework Programme (FP7) of the European Commission. The main objective of FIBRE was to create a common space between the EU and Brazil for Future Internet (FI) experimental research into network infrastructure and distributed applications. Prior to FIBRE, such facilities already were operated, or were being built, by partners in this project from both sides of the Atlantic Ocean. FIBRE was designed so that such a space would enable and encourage closer and more extensive BR-EU cooperation in FI research and experimentation, as well as strengthening the participation of both communities in the increasingly important global collaborations in this important area of network research and development.

The EU-side partners in FIBRE (i2CAT, NWX, NICTA, UnivBRIST, UPMC and UTH) were also participants in the EC's FIRE (Future Internet Research and Experimentation) [6] testbed projects CHANGE [46] [i], OpenLab [47] [ii] and OFELIA [8] An important characteristic of OFELIA was its leveraging of the OpenFlow (OF) approach from Stanford [48]. The participation of OpenLab partners allowed extensions of the OFELIA approach to new testbed environments and use cases not included in OFELIA, especially in the fields of wireless communications. In this latter area, considerable expertise in designing, building and evaluating large-scale testbed systems was brought to the project through the participation of National ICT Australia (NICTA), which has been a major contributor to the development of the OMF control framework [49]. A major objective of the Brazil-EU FIBRE project is the deployment in Brazil of FIBRE2 (Future Internet: BRazilian Experimental Environment), a wide-area network testbed to support user experimentation in the design and validation of new network architectures and applications, which interconnects experimental facilities ("islands") located at the participating institutions (CPqD, RNP, UFF, UFG, UFPA, UFPE, UFRJ, UFSCar, UNIFACS and USP), using RNP's national backbone network [50].

In such a testbed, a high degree of automated resource sharing between experimenters is required, and the testbed itself needed to be instrumented so that precise measurements and accounting of both user and facility resources could be carried out. The Control and Monitoring Framework (CMF) for the FIBRE2 testbed is based on three CMFs developed in the testbed projects OFELIA, ORBIT and ProtoGENI. In order to take best advantage of different testbed functionalities at different sites, FIBRE2 has been created as a federated testbed, which facilitates interoperation with international initiatives. Figure 16 shows diagrammatically the network topology created in the FIBRE project, interconnecting the Brazilian FIBRE2 infrastructure with that of FIBRE's European testbed sites.
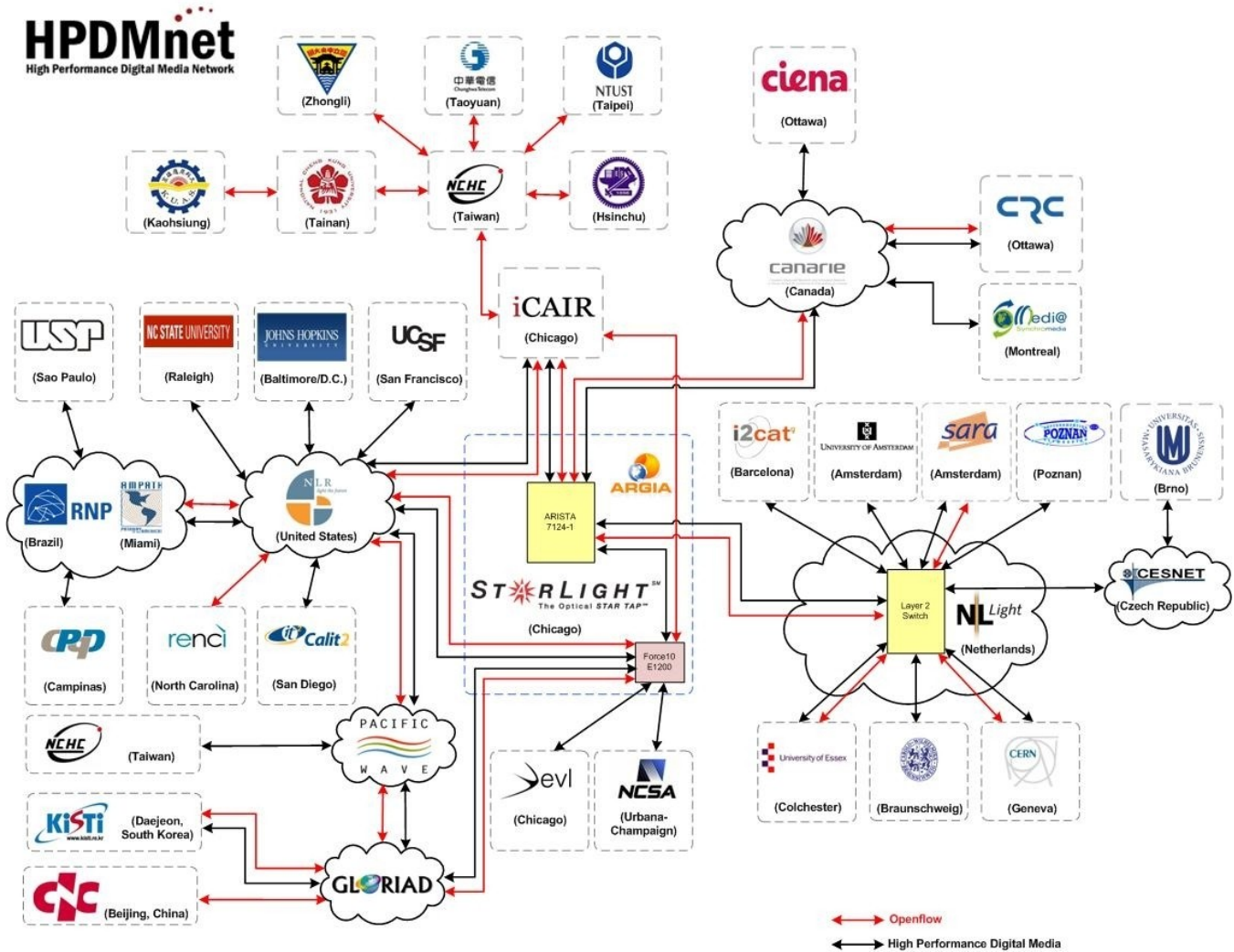
Figure 16: The Federated International Testbed For the FIBRE Project, Including the Brazilian FIBRE2 Component

1.1. High Performance Digital Media Network (HPDMnet)

Questions are often asked about the types of applications that will be supported by SDN. One example is provided by the High Performance Digital Media Network (HPDMnet), which is an international experiment network research testbed that has been used for over ten years to investigate a wide range of topics, including new global scale streaming services, architecture, techniques, and technologies for high performance, ultra high resolution digital media transport based on dynamically programmable L1/L2 paths. [51, 52] Services include those that support high volume digital media streams, required by ultra resolution HD, 4k media, and 8k media. HPDMnet is based on various mechanisms that comprise its control, management and data planes, and the majority of partners in this consortium are moving to implement SDN/OpenFlow capabilities as is shown in Figure 2. [41, 42, 43, 44] (Ref: Figure 17)

Figure 17: The International High Performance Digital Media Network (HPDMnet)

Anticipated Future Services and Resource Expansions

The multiple macro trends toward virtualization (e.g., everything as a service – XaaS), provides for major opportunities for continuing to design, create, and rapidly implement additional – and more innovative – international federated environments for experimental network research. Future plans include the design and implementation of a much wider set of services for such environments, and services that are more accessible, in part, through enhanced APIs with more straightforward authentication and authorization capabilities across multiple domains. The following section describe some of these expansion initiatives. These future testbed environments will be based on international optical networks interconnected next generation multi-service exchanges. [53]

Software Defined Networking Exchanges (SDXs)

As noted, Software-Defined-Networking (SDN) has fundamentally transformed networking services and infrastructure, and it will continue to do so for the foreseeable future. However, current SDN architecture and technologies are single domain oriented, and required capabilities for multi-domain SDN provisioning are fairly challenging. Consequently, the deployment of SDNs has led to multiple isolated SDN "islands." Therefore, the increasing implementations of production SDNs has highlighted the need for the design and creation of Software Defined Networking Exchanges (SDXs). Recently, several research communities have designed and implemented the world's first SDXs, including a prototype at the StarLight facility in Chicago developed by iCAIR and its research partners, initially as a GENI project, and one at the NetherLight facility in

Amsterdam, developed by SURFnet. These SDXs provide multi-domain services enabled by federated controllers and direct data plane connections. International interoperability between the StarLight and NetherLight SDXs was showcased at the TERNENA conferences in May 2014 and June 2015. (Ref: Figure 18)

There are multiple benefits to SDXs: (a) many more options for dynamic provisioning at exchanges, including real time provisioning, (b) faster implementations of new and enhanced services, (c) enabling applications, edge processes and even individuals to directly control core exchange resources, (d) highly granulated views into individual network traffic flows through the exchanges and direct control over those flows (e) enhanced network service and infrastructure management because of those views and (f) substantially improved options for creating customizable network services and infrastructure. The StarLight SDX, which is based on multi-domain services supported by federated controllers and high performance data planes, has been used to demonstrate the potential for creating customized SDXs for specific services and applications, including data intensive domain sciences, especially when based on programmable, segmented, high-capacity 100 Gbps paths. These capabilities, which were showcased by iCAIR and its research partners at SC14 and SC15, included a demonstration of a prototype customized computational bioinformatics SDX.
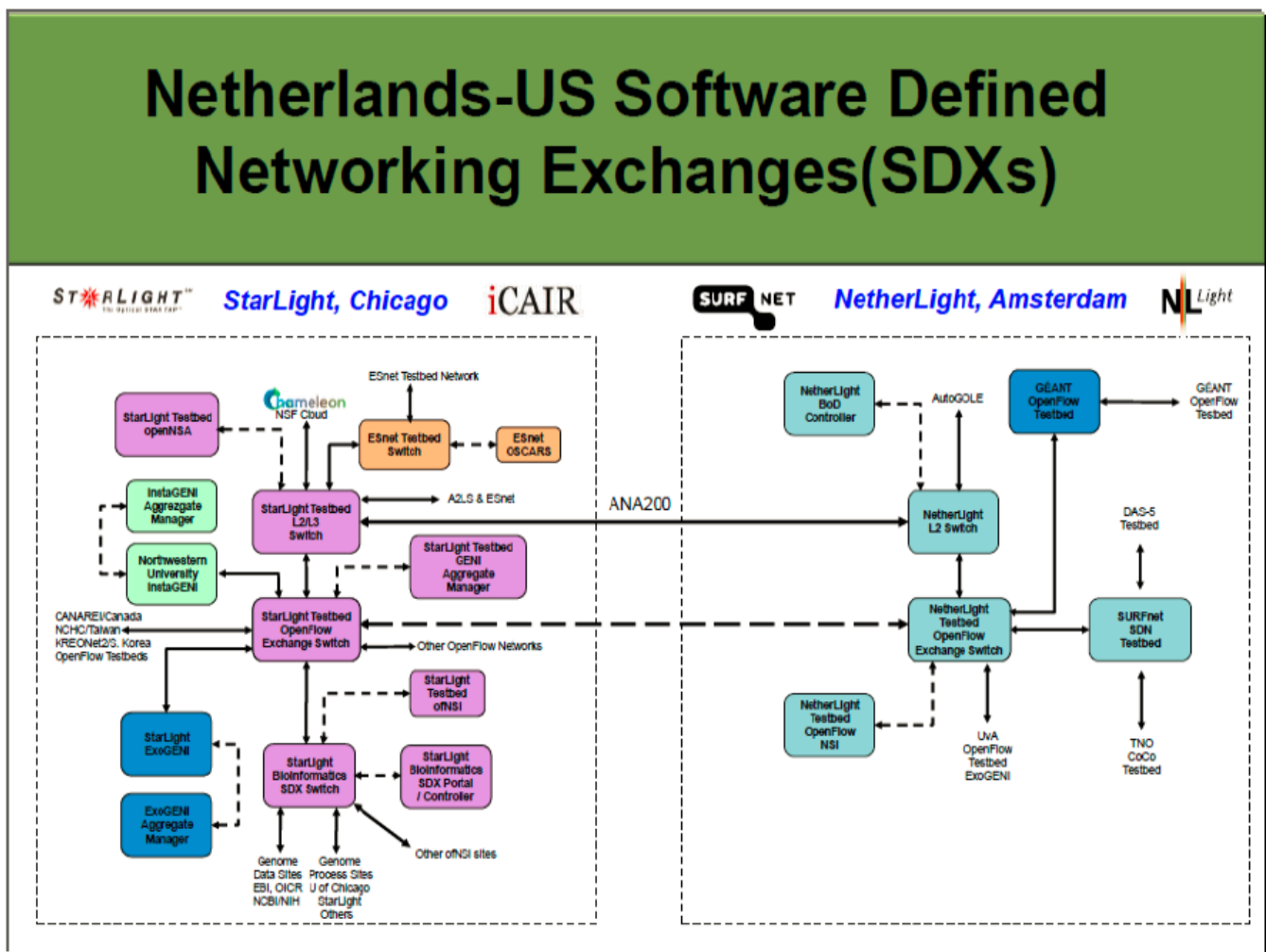

Figure 18: International Interoperability Between StarLight and NetherLight SDXs

Essentially, an SDX is a type of large scale virtual switch, which can provide segmented resources for different domains, locally, nationally or internationally. The substructure for the virtual switch consists of multiple other SDN/OpenFlow switches. [54, 55, 56] (Ref: Figure 19). Recently, iCAIR and the University of Chicago, with multiple international research partners demonstrated how an SDX can be used to create a virtual exchange customized to support the complex workflows

required to optimally support new techniques for precision medicine – precision medicine enabled by precision networking. iCAIR also established a project with the University of Massachusetts to explore how SDXs could be used to support new techniques for weather prediction and visualization.
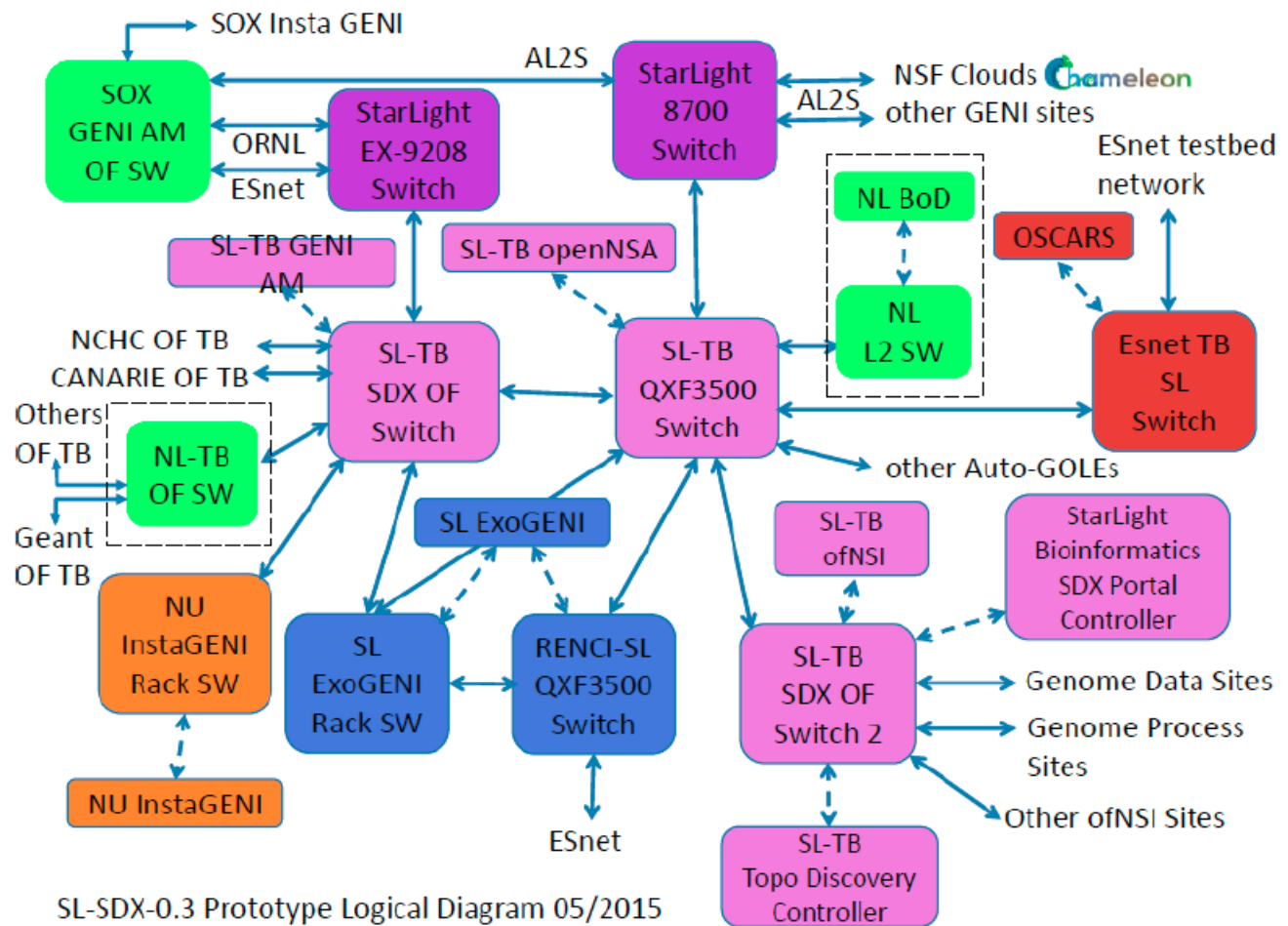

SL-SDX-0.3 Prototype Logical Diagram 05/2015

Figure 19: Logical Diagram of StarLight SDX

Software Defined Infrastructure (SDI) and Cloud Testbed Integration

Other areas being investigated include SDX extensions to Software Defined Infrastructure (SDI) integrating additional resources including compute facilities, clouds, Grids, storage, instruments, mobile devices, sensors, and other resources. In general, a major trend has been seen toward the incorporation into these environments of additional types of highly programmable resources (at an extremely granulated level), e.g., compute clouds, specialized compute devices such as those based on GPUs and FPGAs, storage systems, instrumentation, wireless fabrics, RFP based sensors, edge devices etc. At this time, almost all sites have similar sets of core capabilities. However, in the future different sites may specialize in highly differentiated services and resources, such as those that may specialize on sensor networks. Another trend has been the development of many more additional research resource sites at many places around the world.

In part, this area is being initially investigated through an integration with one of the NSFCloud testbeds. For example, the Chameleon distributed cloud testbed [57, 58] has been integrated with GENI, as has the companion project, CloudLab. Related to these projects is ongoing research with the University of Tokyo and other partners in Japan on architecture and technologies for Distributed Slice Exchanges (DSEs), which closely integrate distributed environments across multiple international multi-domain sites.

Emerging Architecture and Design Trends for Anticipated Future Facilities

Based on existing design trends and emerging architecture for experimental research facilities, a number of premises can be extrapolated about future developments of such environments. As noted, the macro trends toward virtualization at all levels – proving every resource as a service (XaaS) is leading to an explosive growth in such services. Currently, it is possible to create "service factories," based on large scale virtualization capabilities, using as a foundation a rich array of programmable network middleware and a wide array of underlying infrastructure, which together become a platform for innovation. These platforms will continue to evolve rapidly, just as Grid platforms evolved to incorporate programmable networking architecture and technology, a progression that did much to inform today's programmable networking t.[59, 60]

A key attribute of emerging and future environments is that they allow not just large organizations to create, deploy and operate networks but also they enable individuals to create their own large scale networks, customizing them to meet individualized, specialized requirements vs general requirements. The potential for these capabilities is accelerating because the underlying core infrastructure is virtualized at the same time that it is rapidly declining in cost as component technologies move to commodity.

Conclusions

This chapter provides an overview of the international capabilities of GENI and related network research environments, with descriptions of the services required by research communities, basic architectural approaches, existing services and facilities, and examples of current research experiments being conducted within these environments. The chapter also anticipates emerging architecture and design trends for anticipated future services for such international experimental network environments, as well as facilities, and expansions to many additional resources. Key macro trends are those that enable virtual and physical network resource to be abstracted, so that customized resource slices across international WANs can be dynamically created and implemented. Such contiguous integrations of highly distributed resourcescan be manipulated using new types of orchestration techniques, which are being made available not only to systems operators, but also to edge applications, processes and individuals.

Acknowledgments

References

[1] M. Berman, J. Chase, L Landweber, A. Nakao, M. Ott, D. Raychaudhuri, R. Ricci g, I. Seskar, GENI: A Federated Testbed for Innovative Network Experiments, Special issue on Future Internet Testbeds Computer Networks Vol 61 Mar. 14, 2014, pp. 5-23.

[2] R. Strijkers, M. X. Makkes, C. de Laat, R. Meijer, "Internet Factories: Creating Application-Specific Networks On-Demand", Special Issue On Cloud Networking and Communications, Computer Networks, Elsevier Journal of Computer Networks, http://dx.doi.org/10.1016/j.comnet.2014.01.009.

[3] http://www.ogf.org/gf/group_info

[4] P. Mell, T. Grance, Definition of Cloud Computing, Special Publication 800-145, Recommendations of the National Institute of Standards and Technology (NIST), www.nist.gov/itl/cloud

[5] National Science Board, International Science and Engineering Partnerships: A Priority for U.S. Foreign Policy and Our Nation's Innovation Enterprise, National Science Foundation, February 14, 2008.

[6] J. Jofre, C. Velayos, G. Landi, M. Giertych, A. Humed, G. Francis d, A. Oton, Federation of the BonFIRE Multi-Cloud Infrastructure With Networking Facilities, Special Issue On Future Internet Testbeds Computer Networks Vol 61 Mar. 14, 2014, pp. 184-196.

[7] http://www.ist-phosphorus.eu/

[8] M. Suñé, L. Bergesio, H. Woesner, T. Rothe, A. Köpsel, D. Colle, B. Puype, D. Simeonidou, R. Nejabati, M. Channegowda, M. Kind, T. Dietz, A. Autenrieth, V. Kotronis k, E. Salvadori h, S. Salsano i, M. Körner j, S. Sharma, Design and Implementation of the OFELIA FP7 Facility: The European OpenFlow Testbed, Special Issue On Future Internet Testbeds, Computer Networks Vol 61 Mar. 14, 2014, pp 132-150

[9] N. McKeown, et al.: OpenFlow: Enabling Innovation in Campus Networks. ACM, SIGCOMM Computer Communication Review 38(2), 2008, pp. 69–74.

[10] B. Belter, J. Martinez, J. Aznar, J. Riera, L. Contreras, M. Lewandowska, M. Biancani, J. Buysse, C. Develder, Y. Demchenko, P. Donadio, D. Simeonidou i, R. Nejabati, S. Peng, L. Drzewiecki, E. Escalona, J. Espin, The GEYSERS Optical Testbed: A Platform For The Integration, Validation and Demonstration Of Cloud-Based Infrastructure Services, Special issue On Future Internet Testbeds Computer Networks Vol 61 Mar. 14, 2014, pp 197-216

[11] M. Campanella, F. Farina, The FEDERICA Infrastructure and Experience, Special issue On Future Internet Testbeds Computer Networks Vol 61 Mar. 14, 2014, pp. 176-183

[12] D. Schwerdel, B. Reuther, T. Zinner, P. Müller, P. Tran-Gia, Future Internet Research and Experimentation: The G-Lab Approach Special issue On Future Internet Testbeds Computer Networks Vol 61 Mar. 14, 2014, pp. 102-117.

[13] www.fed4fire.eu

[14] S.Thorpe, L. Battestilli, G. Karmous-Edwards, A. Hutanu, J. MacLaren, J. Mambretti, J. Moore, K. Sundar, Y. Xin, Yufeng, A. Takefusa, M. Hayashi, A. Hirano, S. Okamoto, T. Kudoh, T. Miyamoto, Y. Tsukishima, T. Otani, H. Nakada, H. Tanaka, A. Taniguchi, Y. Sameshima, M. Jinno, "G-Lambda and EnLIGHTened: Wrapped in Middleware Co-Allocating Compute and Network Resources Across Japan and the US," Proceedings of the First International Conference on Networks for Grid Applications ,Lyon, France, SIGARCH: ACM Special Interest Group on Computer Architecture, Published by Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering (ICST) Brussels, Belgium, 2007.

[15] D. Kim, J. Kim, G. Wang, J-H Park, S-H Kim, K-GENI Testbed Deployment and Federated Meta Operations Experiment over GENI and KREONET, Special issue on Future Internet Testbeds Computer Networks Vol 61 March 14, 2014, pp. 39-50.

[16] M. Stanton, RNP Experiences and Expectations in Future Internet Research and Development, New Network Architectures, Studies in Computational Intelligence Volume 297, 2010, pp 153-166.

[17] www.savinetwork.ca

[18] J. van der Ham, J. Stéger, S. Laki, Y. Kryftis, V. Maglaris, C. de Laat, "The NOVI Information Models", Future Generation Computer Systems, Available online 18 December 2013, ISSN 0167-739X, http://dx.doi.org/10.1016/j.future.2013.12.017.

[19] J..J. van der Ham, F. Dijkstra, F. Travostino, H.M.A. Andree and C.T.A.M. de Laat, "Using RDF to Describe Networks", iGrid2005 special issue, Future Generation Computer Systems, volume 22 issue 8, pp. 862-867 (2006).

[20] M Ghijsen, J van der Ham, P Grosso, C Dumitru, H Zhu, Z Zhao, C de Laat, "A Semantic-Web Approach for Modeling Computing Infrastructures", Journal of Computers and Electrical Engineering, Elsevier, Volume 39, Issue 8, November 2013, Pages 2553–2565, doi: 10.1016/j.compeleceng.2013.08.011.

[21] I. Baldine I, Y. Xin, A. Mandal, C. Heermann, J. Chase, V. Marupadi, et al. Networked Cloud Orchestration: a GENI Perspective. Workshop On Management Of Emerging Networks and Services; 2010.

[22] Global Lambda Integrated Facility (GLIF) www.glif.is

[23] Global Ring Network For Advanced Applications Development (GLORIAD), www.gloriad.org

[24] G. Roberts, T. Kudoh, I. Monga, J. Sobieski, J. MacAuley, C. Guok, NSI Connection Service V2.0, Open Grid Forum, GWD-R-P, NSI-WG 2013.

[25] http://groups.geni.net/geni/wiki/IGENI

[26] J. Mambretti, J. Chen, F. Yeh, International Network Research Testbed Facilities Based on OpenFlow: Architecture, Services, Technologies, and Distributed Infrastructure Proceedings, 18th IEEE International Conference on Networks (ICON), 2012, pp. 234 – 242.

[27] A. Bavier, M. Yuen, J. Blaine, R. McGeer, A.Young, Y Coady, C. Matthews,C. Pearson, A. Snoeren, J. Mambretti: TransCloud - Design Considerations for a High-performance Cloud Architecture Across Multiple Administrative Domains. CLOSER 2011, pp. 120-126.

[28] http://tomato-lab.org

[29] D. Schwerdel, D. Hock, D. Günther, B. Reuther, P. Müller, and P. Tran-Gia. ToMaTo - a network experimentation tool. In 7th International ICST Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TridentCom 2011), Shanghai, China, April 2011.

[30] D. Schwerdel, B. Reuther, T. Zinner, P. Müller and P. Tran-Gia. Future Internet Research and Experimentation: The G-Lab Approach. Computer Networks (2014), DOI information: 10.1016/j.bjp.2013.12.023.

[31] J. Mambretti, J. Chen, F. Yeh, T-L. Liu, M-Y. Luo, C-S.Yang, R. van der Pol, S. Boele, F. Dijkstra, A. Barczyk, G. van Malensteinz, Openflow Services For Science: An International Experimental Research Network Demonstrating

Multi-Domain Automatic Network Topology Discovery, Direct Dynamic Path Provisioning Using Edge Signaling And Control, Integration With Multipathing Using MPTCP, 2012 SC Companion: High-Performance Computing, Networking, Storage and Analysis (SCC); Nov. 2012

[32] R. van der Pol, S. Boele, F. Dijkstra, A. Barczyk, G. van Malenstein, J.H. Chen, J. Mambretti, Multipathing with MPTCP and OpenFlow, High Performance Computing, Networking, Storage and Analysis (SCC), 2012 SC Companion, Nov 2012

[33] W-Y. Huang, J-W. Hu, S-C. Lin, T-L. Liu, P-W. Tsai, C-S. Yang, F. Yeh, J. Hao Chen, J. Mambretti, "Design and Implementation of An Automatic Network Topology Discovery System for the Future Internet Across Different Domains," Proceedings of IEEE 26th International Conference on Advanced Information Networking and Applications Workshops (AINAW'12), Singapore, March 2012.

[34] M-Y Luo S-W Lin, and J-Y Chen, From Monolithic Systems to a Federated E-Learning Cloud System, IEEE International Conference on Cloud Engineering, March 25-28, San Francisco, California, 2013.

[35] P-W Tsai, P-W Cheng, M-Y Luo, Te-Lung Liu and Chu-Sing Yang, Planning and Implantation of NetFPGA Platform on Network Emulation Testbed, Proc. Asia Pacific Advanced Network, Network Research Workshop, Delhi, India, August 22, 2011.

[36] Testbed@TWISC, http://testbed.ncku.edu.tw/

[37] M-Y Luo, J. Chen, J. Mambretti, S-W Lin, P-W Tsai, Fei Yeh, and Chu-Sing Yang, Network Virtualization Implementation Over Global Research Production Networks, Journal of Internet Technology, Vol. 14, No.7, 2013, pp. 1061-1072.

[38] M-Y Luo, J-Y Chen, "Towards Network Virtualization Management for Federated Cloud Systems", IEEE 6th International Conference on Cloud Computing, June 27-July 2, 2013, Santa Clara, CA, USA.

[39] M-Y Luo, J-Y Chen, "Software Defined Networking across Distributed Datacenters over Cloud," 5th IEEE International Conference on Cloud Computing Technology and Science (IEEE CloudCom), Bristol, UK, December 2-5, 2013.

[40] R. Koning, P. Grosso and C. de Laat, "Using Ontologies For Resource Description in the CineGrid Exchange", Future Generation Computer Systems, Volume 27, Issue 7, June 2011, Pages 960-965.

[41] NOVI – Networking Innovations Over Virtualized Infrastructures. <http://www.fp7-novi.eu/>.

[42] B. Ahlgren, C. Dannewitz, C. Imbrenda; D. Kutscher, B. Ohlman, A Survey of Information-Centric Networking, Communications Magazine, IEEE , vol.50, no.7, pp.26,36, July 2012

[43] M. J. Reed, Traffic Engineering For Information-Centric Networks, Communications (ICC), 2012 IEEE International Conference on , vol., no., pp.2660,2665, 10-15 June 2012

[44] D. Trossen, G. Parisis, "Designing and realizing an information-centric Internet," Commun. Mag. IEEE, Vol 50, no. 7, July, pp. 60–67, 2012.

[45] http://www.fibre-ict.eu

[46] http://www.change-project.eu

[47] http://www.ict-openlab.eu

[48] N. McKeown et al., OpenFlow: Enabling Innovation in Campus Networks, ACM SIGCOMM Computer Communication Review, 2008, 2, 69-74.

[49] T. Rakotoarivelo, M. Ott, G. Jourjon, I. Seskar, OMF: A Control and Management Framework for Networking Testbeds, ACM SIGOPS Operating Systems Review, 2010, 4.

[50] A. Abelem, et al., FIT@BR - a Future Internet Testbed in Brazil, Proceedings of the Asia-Pacific Advanced Network, v. 36, p. 1-8, 2013.

[51] J. Mambretti, M. Lemay, S. Campbell, H. Guy, T. Tam, E. Bernier, B. Ho, M. Savoie, C. de Laat, R. van der Pol, J. Chen, F. Yeh, S. Figuerola, P. Minoves, D. Simeonidou, E. Escalona, N. Amaya Gonzalez, A. Jukan, W. Bziuk, D. Kim, K-J Cho, H-L Lee and T-L Liu, "High Performance Digital Media Network (HPDMnet): An Advanced International Research Initiative and Global Experimental Testbed.", Future Generation Computer Systems, Volume 27, Issue 7, June 2011, pp. 893-905.

[52] A. Jukan, J. Mambretti, "Evolution of Optical Networking Toward Rich Digital Media Services" Proceedings of the IEEE 100(4), 2012, pp. 855-871.

[53] J. Mambretti, J. Chen, F. Yeh, Creating Environments For Innovation: Designing and Implementing Advanced Experimental Network Research Testbeds Based On The Global Lambda Integrated Facility and the StarLight Exchange, Special issue On Future Internet Testbeds Computer Networks Vol 61 Mar. 14, 2014, pp. 118-131

[54] N. Feamster, J. Rexford, S. Shenkerz, D. Levin, R. Clark, J. Bailey, SDX: A Software Defined Internet Exchange, White Paper, University of Maryland.

[55] J. Mambretti, J. Chen, F. Yeh, Software-Defined Network Exchanges (SDXs): Enabling Capabilities for Distributed Clouds With SDN Multi-Domain and Multi-Services Techniques, Accepted, Workshop on Future Internet Testbeds and

Distributed Clouds (FIDC), co-located with the International Teletraffic Congress in Karlskrona, Sweden, Sept 9-11, 2014.

[56] J. Mambretti, J. Chen, F. Yeh , Software-Defined Network Exchanges (SDXs) and Infrastructure (SDI): Emerging Innovations In SDN and SDI Interdomain Multi-Layer Services and Capabilities , First International Science and Technology Conference: Modern Networking Technology: SDN and NFV – The Next Generation of Computational Infrastructure, Moscow, Russia, October 28-29, 2014, Published By IEEE, Science and Technology Conference (Modern Networking Technologies) (MoNeTeC), 2014 International.

[57] J. Mambrretti, J. Chen, F. Yeh, Next Generation Clouds, The Chameleon Cloud Testbed, and Software Defined Networking (SDN), Proceedings, International Conference on Cloud Computing Research and Innovation (ICCCRI 2015), 26-27 October 2015, Singapore (forthcoming).

[58] www.chameleoncloud.org

[59] F. Travostino, J. Mambretti, G. Karmous-Edwards (editors), Grid Networks: Enabling Grids with Advanced Communication Technology, John Wiley & Sons, July 2006.

[60] T. Doulamis, J. Mambretti, I. Tomkos, D. Varvarigou, Ed. Networks For Grid Applications. Third International ICST Conference, GridNets 2009, Athens, Greece, September 8–9, 2009. Revised Selected Papers