A Model Driven Intelligent Orchestration Approach to Service Automation in Large Distributed Infrastructures

AI-Science'18: Autonomous Infrastructure for Science

Tempe, AZ, United States

June 11, 2018

Xi Yang
University of Maryland

Raj Kettimuthu
Argonne National Laboratory

Eun-Sung Jung
Hongik University

Tom Lehman
University of Maryland

Linda Winkler
Argonne National Laboratory

Research Project Sponsors

- **Resource Aware Intelligent Network Services(RAINS)**

U.S. DEPARTMENT OF ENERGY
Office of Science

  - University of Maryland/Mid-Atlantic Crossroads (UMD/MAX)
  - Argonne National Laboratory (ANL)

- **Regional Embedded Cloud for As-a-Service Transformation (RECAST)**

NSF

  - University of Maryland/Mid-Atlantic Crossroads (UMD/MAX)

Vision and Motivation

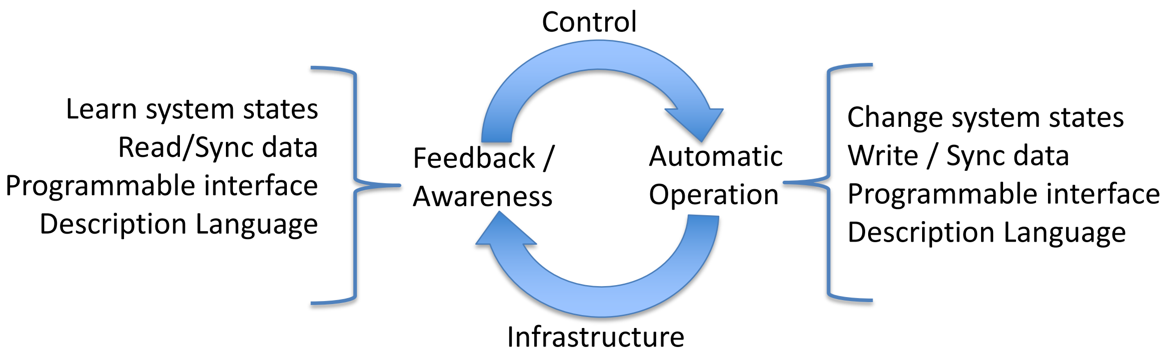
- Users are looking for integrated services through consolidated interfaces and/or portals
- The desire is to eliminate the need for manual and human-in-the-loop integration of compute, storage, and network resources.
- That is, we have not solved the end-to-end problem in a multi-resource context
- A key problem is that compute, storage, instrument, and network resources are not “integrated” with each other or with domain application workflows.
- An infrastructure for topology and service awareness is needed

Approach and Solution

- **Multi-Resource Orchestration:** integrating and orchestrating the network and network services with the things that attach to the network – compute, storage, clouds, and instruments.
- **Model Driven:** using models to describe resources in order to allow integrated reasoning, abstraction, and user centered services
- **Intelligent Computation Services:** Model driven, multi-resource computation services to enable orchestration services in response to high level user requests.
- We want to “Orchestrate the Automaters”

Why Modeling

- **Automation and Orchestration in a control feedback loop**



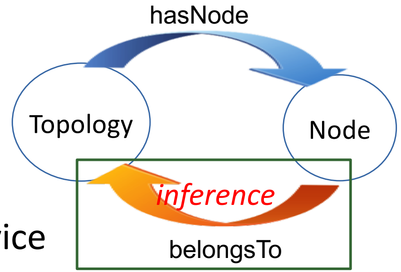
- **Can do all of these via modeling:**
  - **States/Data:** Uniform modeling for all resources and services through "linked data" standards
  - **Interface:** Model carries all syntax and semantics, which requires only very slim interface for transport
  - **Language:** Model is a language that can represent any concepts, relationships and workflows

Model Driven Orchestration

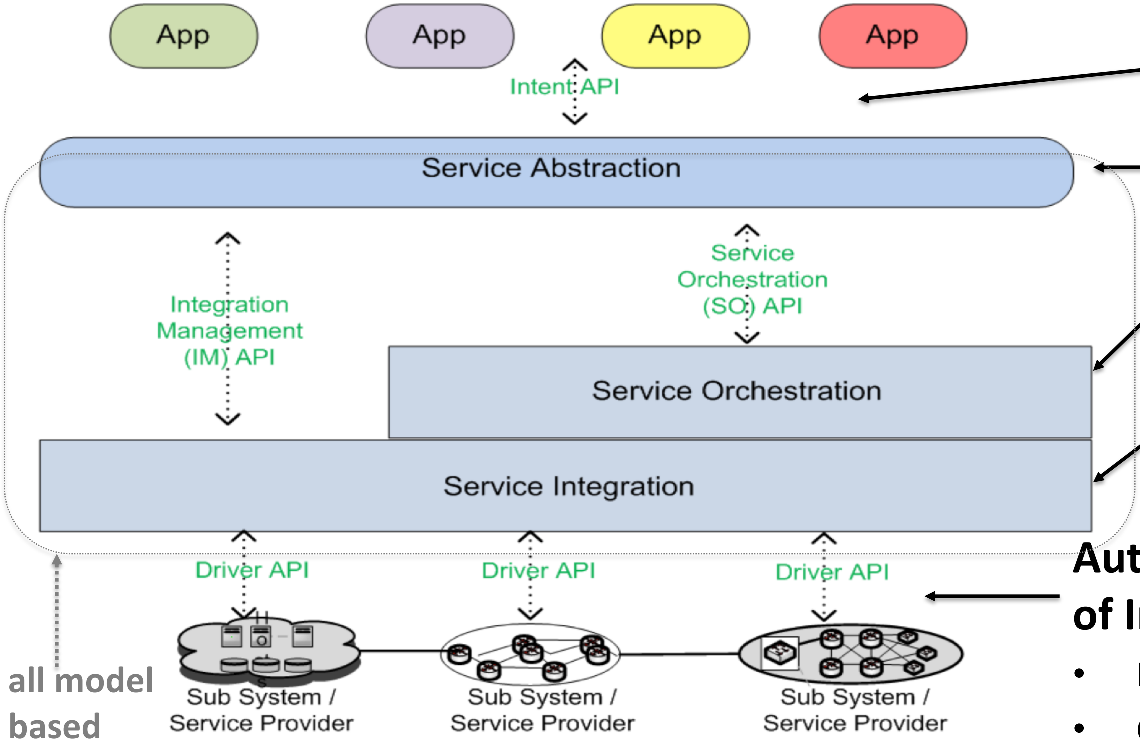
- ESnet and others in the R&E Network community are using Network Markup Language (NML) to describe network resources
  - OGF GFD-R-P.206, NML Base Schema version 1
  - Used as part of OGF Network Service Interface (NSI)
- RAINS extended NML to cover multi-resources, so that we can integrate the network and the resources attached to the network (Science DMZs, DTNs, private clouds, public clouds, instruments, storage systems)
- Multi-Resource Markup Language (MRML)
  - https://github.com/MAX-UMD/nml-mrml

Modeling Framework

- **Standards based framework and tools**
  - Built on established Semantic Web, a.k.a. Linked Data, standards.
  - Resource Description Framework (RDF) - define ways for how to model (syntax)
  - Web Ontology Language (OWL) - define ways for what to model (semantics)
  - Leverage open source programmatic framework (Apache Jena), resource query (SPARQL) and ontology reasoning engines.
  - Represented as collection of subject-predicate-object (triples)
- **Multi-resource service modeling**
  - Resources: Topology, Node, Port, Link, Label, Group etc.
  - Services: SwitchingService, RoutingService, HypervisorService, StorageService etc.
  - Relations: How Resources and Services are connected and related.
  - Support arbitrary hierarchy, virtualization and abstraction
  - Composable, extensible and scalable



Service-Oriented Modeling Stack



all model based processing

Abstract Intent based requests

Intent request to model conversion

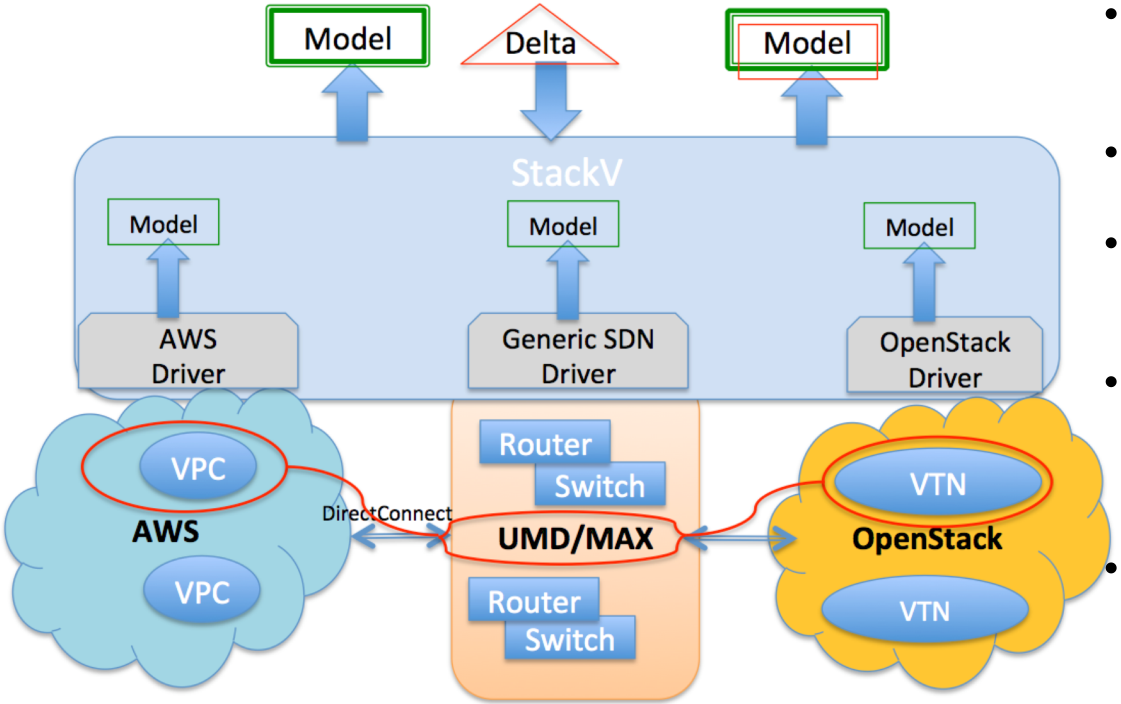
Service Workflow processing

Integration of multi-resource/provider models

Automatic Model Generation of Infrastructure and Services

- **Resource Owner Generated or**
- **Orchestration System Generated (via existing APIs)**

Model Driven Integration and Instantiation



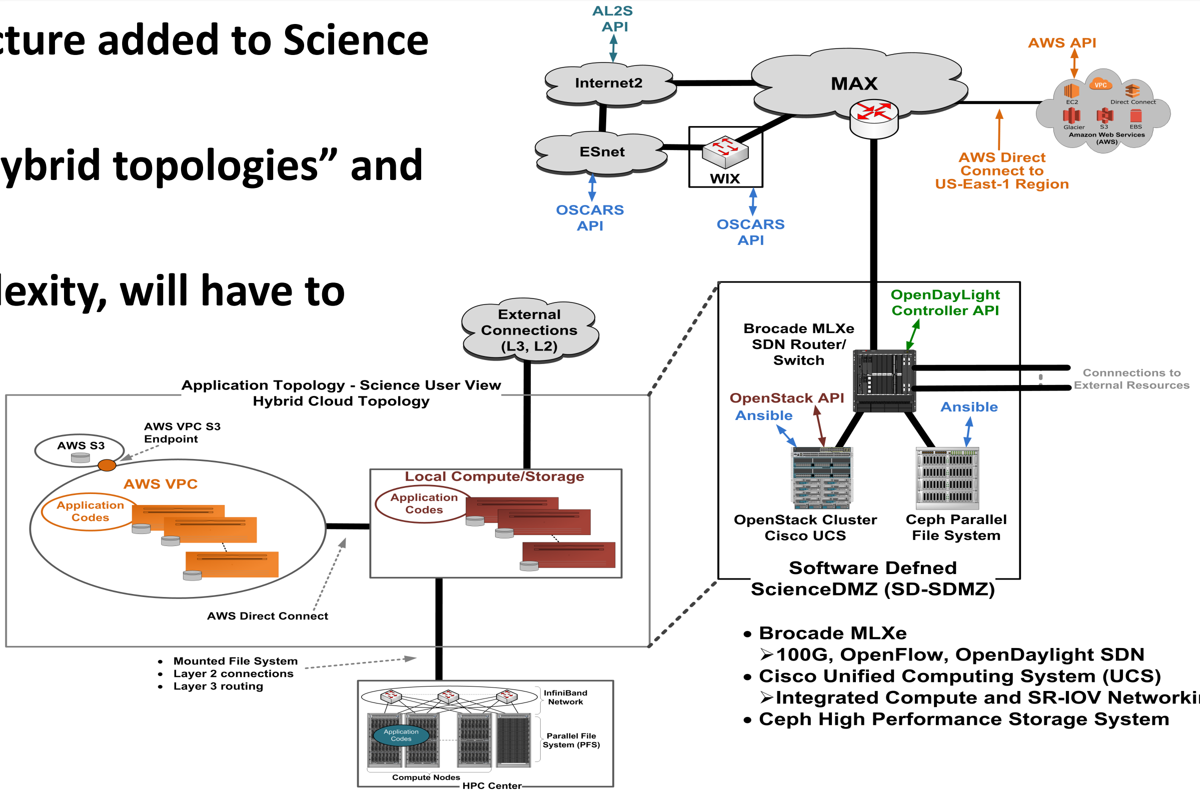
- **Start with an infrastructure model**
- **Process User Request**
- **Construct a "Delta" to the infrastructure model**
- **StackV pushes this "Delta" to the appropriate Resource/Driver**
- **New infrastructure Model is returned which reflects service provisioning**

Current Resource Modeling

- **Science DMZs**
  - Traditional Bare Metal based Data Transfer Nodes (DTNs)
  - Software Defined Science DMZ (SD-SDMZ)
    - Compute (virtual machines), Storage, Networking
- **Software Defined Exchanges (SDX)**
- **Public Clouds (AWS)**
- **Private Clouds (OpenStack)**
- **The interconnecting networks**
  - Regional networks, Internet2, ESnet

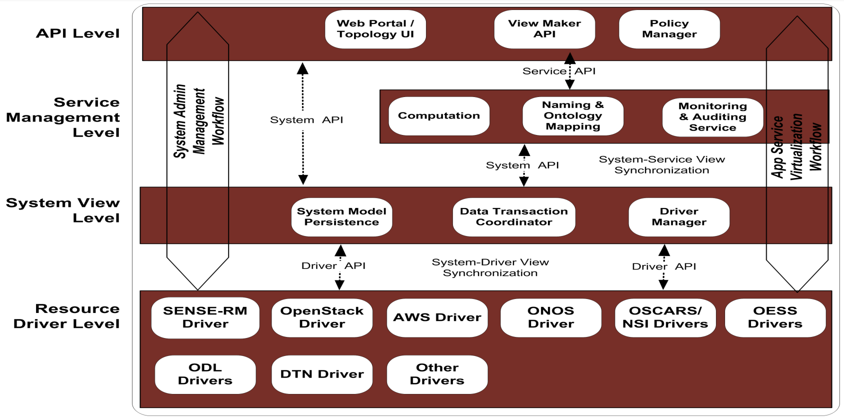
UMD/MAX SD-SDMZ

- Virtualized infrastructure added to Science DMZ
- Now we can build “hybrid topologies” and custom services
- Added a lot of complexity, will have to automate
- Also want to “orchestrate” external resources like AL2S, OSCARS networks



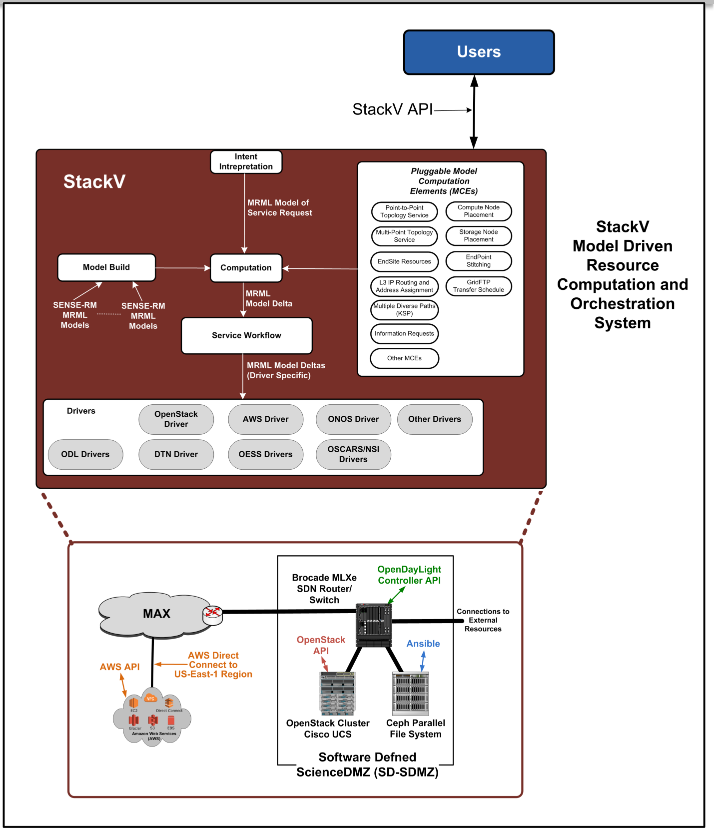
- Brocade MLXe
  - ➢100G, OpenFlow, OpenDaylight SDN
- Cisco Unified Computing System (UCS)
  - ➢Integrated Compute and SR-IOV Networking
- Ceph High Performance Storage System

StackV - Orchestration Suite



Architecture

Conceptual View

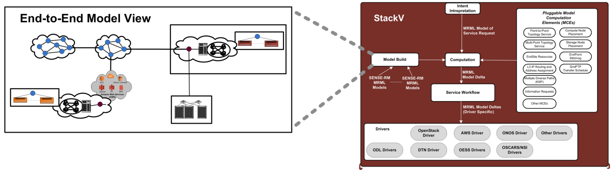


- UMD/MAX developed StackV is an open source (geni public license) model driven orchestration system:
  - github.com/MAX-UMD/stackV.community
- Would like to build an open source community around, available for others to use as desired

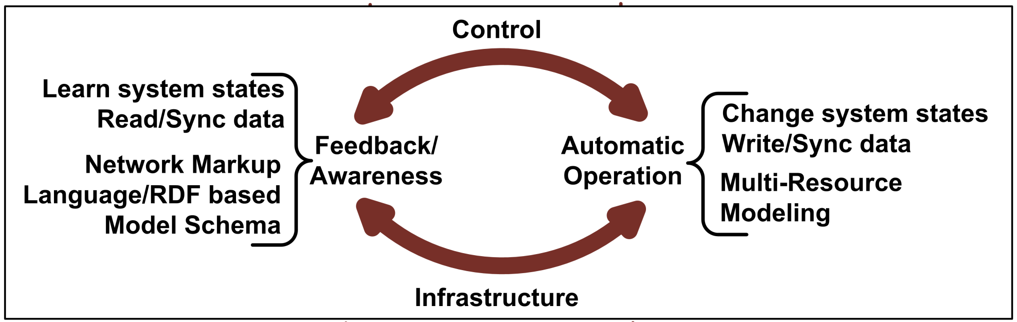
Model Based Control and Orchestration

StackV Design Principles

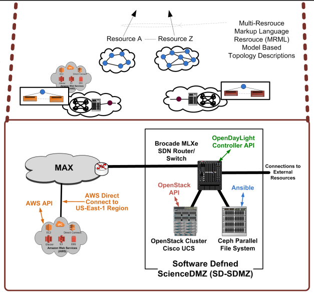
- Focus is turnkey services which can be customized for different users
- Model based allows for rapid integration of new resource types
- Modular Computation Elements facilitate custom service workflow construction
- A DevOps model for service construction
- Generic control of infrastructure for custom construction of user facing services



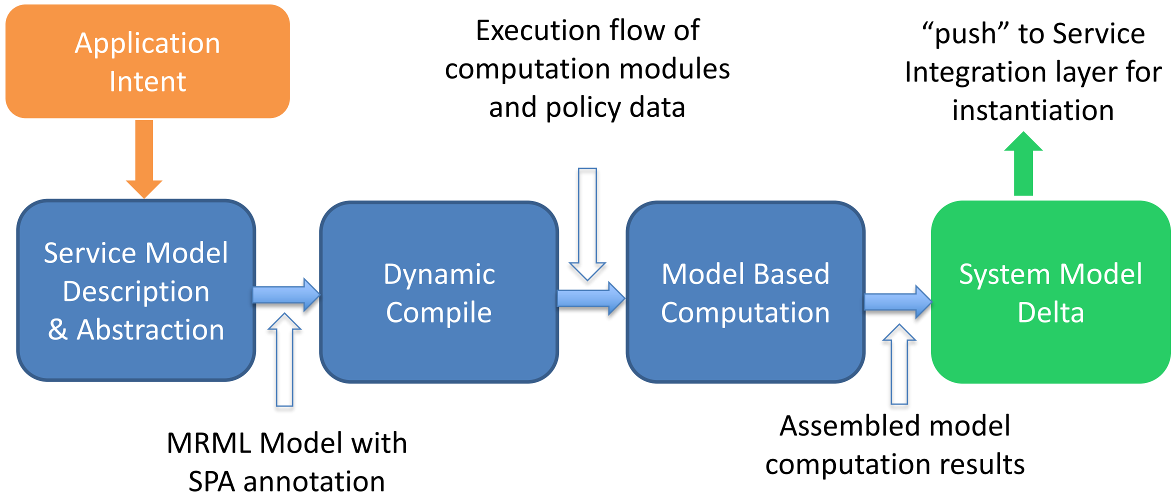
Allow the machines to automate, iterate, react, and adjust to find solutions and not bring the humans in until absolutely necessary



Automated Infrastructure Models created using available System APIs



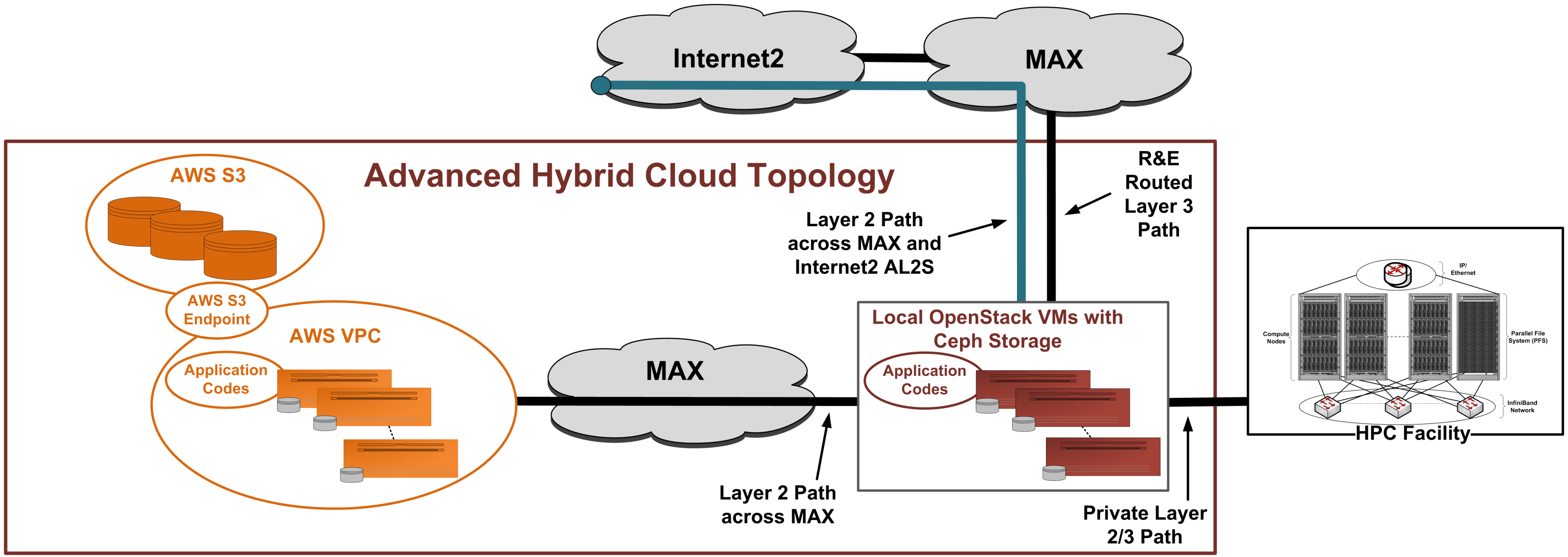
Service Orchestration Workflow



Simple Policy Annotation (SPA) - defines resource inter-dependency, workflow order and service chaining properties for use intent based services

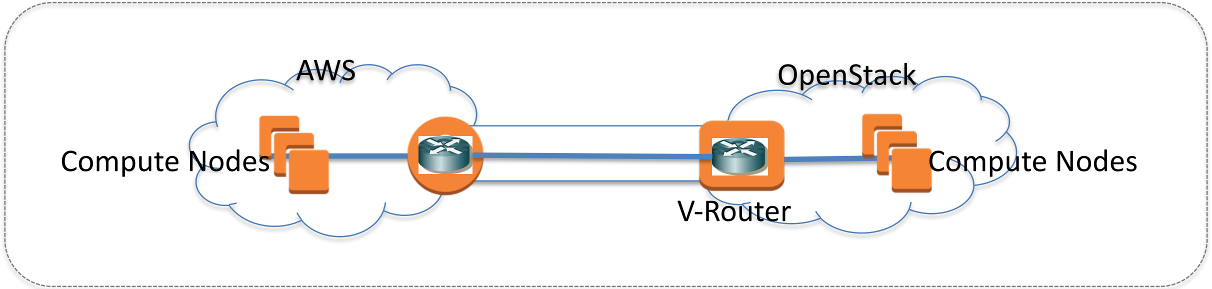
Hybrid Cloud Inter-Networking (HCIN)

Example Topology

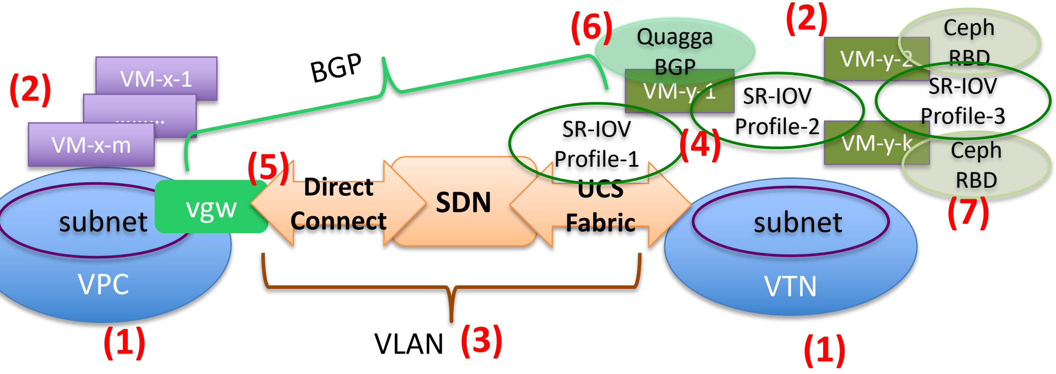


Model Computation Workflow Illustration

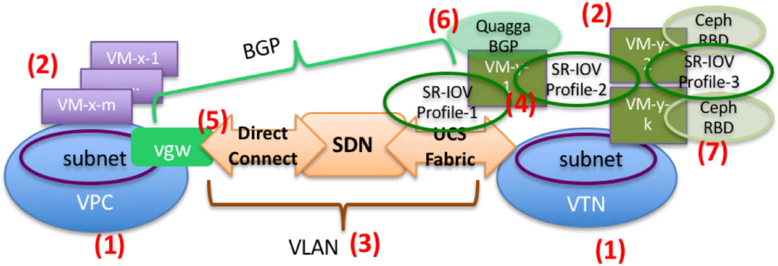
User Intent Request



Orchestrated Workflow

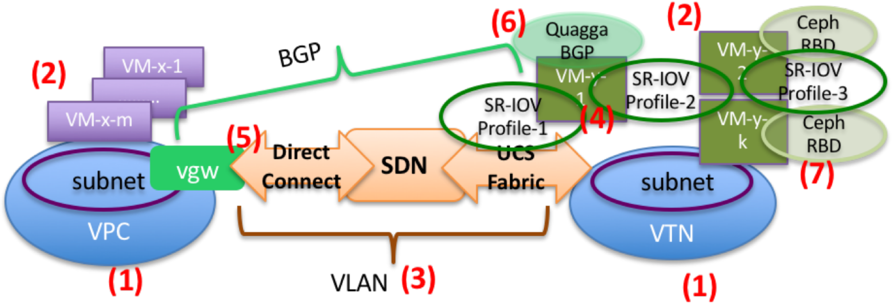


Model Computation Workflow Illustration



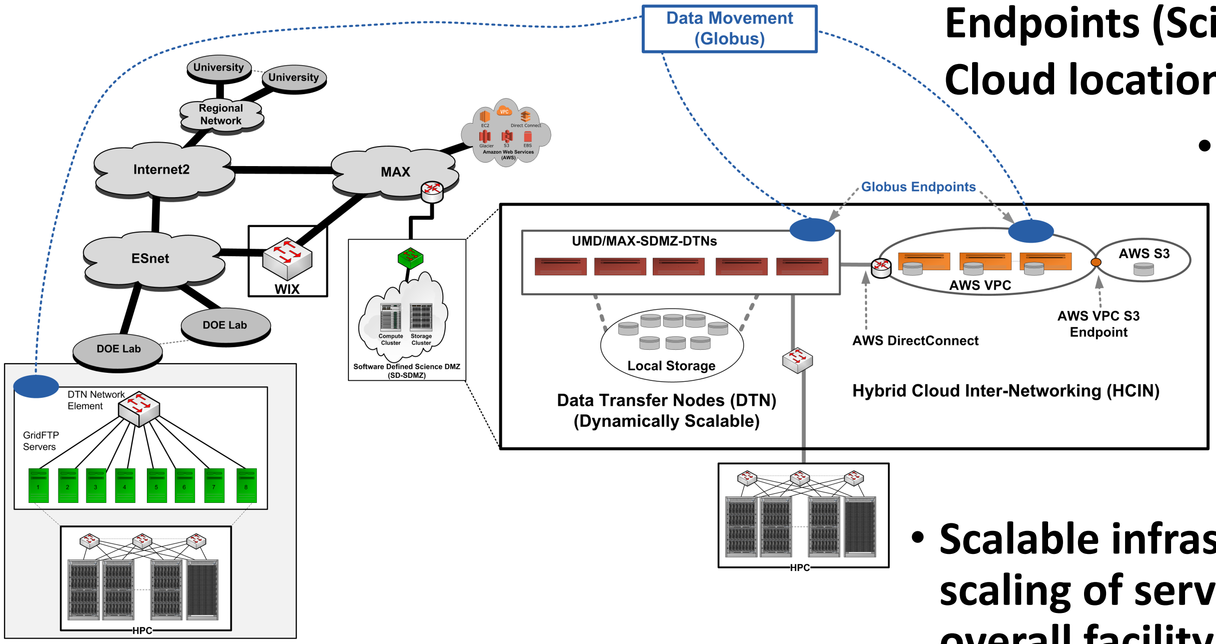
1. Virtual_Cloud_Network MCEs compute model statements for adding AWS and OpenStack virtual networks, subnets, gateways and routes.
2. VM_Placement MCEs add statements to place VMs into specific AWS subnets or to OpenStack subnets and hosts.
3. L2_Path_Computation MCE computes layer2 path and add statements for all the VLANs on the switches, DirectConnect and UCS fabric between AWS VPC and OpenStack hosts from 1 and 2.
4. SRIOV_Stitching uses the results of 2 and 3 to add model statements for OpenStack VM SR-IOV interfaces and for their connectivity to layer2 VLAN path.

Model Computation Workflow Illustration



5. In parallel to 4, DirectConnect_Stitching uses the results of 2 and 3 to to add statements for AWS VPC to attach to a DirectConnect VLAN and then to the layer2 path.
6. NFV_Quagga_BGP MCE adds model statements for deploying Quagga based virtual router on an OpenStack VM Linux using the results of 4 and 5.
7. Networked_Block_Storage MCE adds model statements for creating Ceph Block Devices (RBD) and attaching and mounting to designated VMs with traffic routed to SR-IOV interfaces. This depends on 2, 4 and 6.

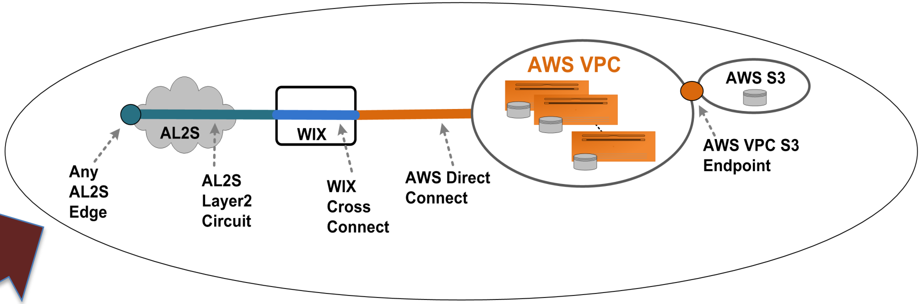
Service Topology for Data Transfer

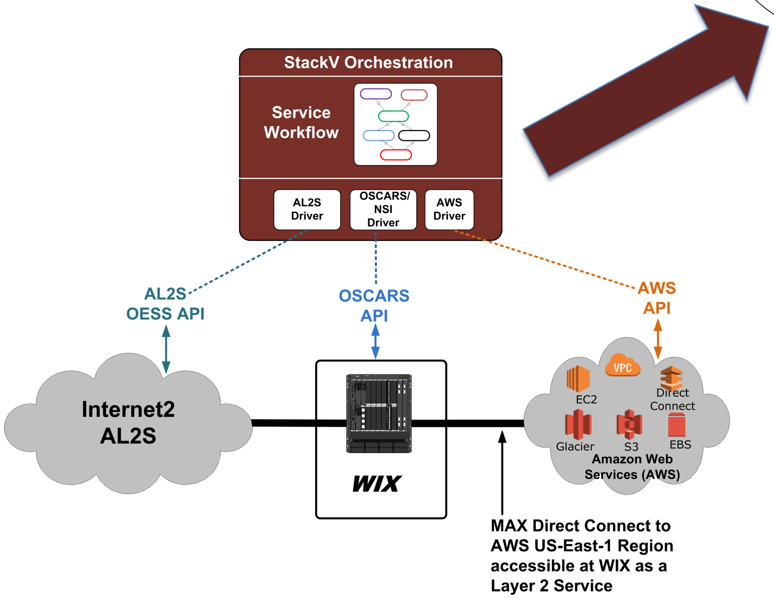


- Automated DTN Creation with Globus Endpoints (Science DMZ and Public Cloud locations)
- Dedicated per User (SD-SDMZ infrastructure resources released for others after work is complete)
- Scalable infrastructure (dynamic scaling of service resources based on overall facility load and expected activity)

EdgeCloud Connection Service

Separate StackV instance for this service





Orchestrates AL2S, WIX, and AWS Direct Connect provisioning

SENSE Project Utilizing RAINS Modeling and Orchestration

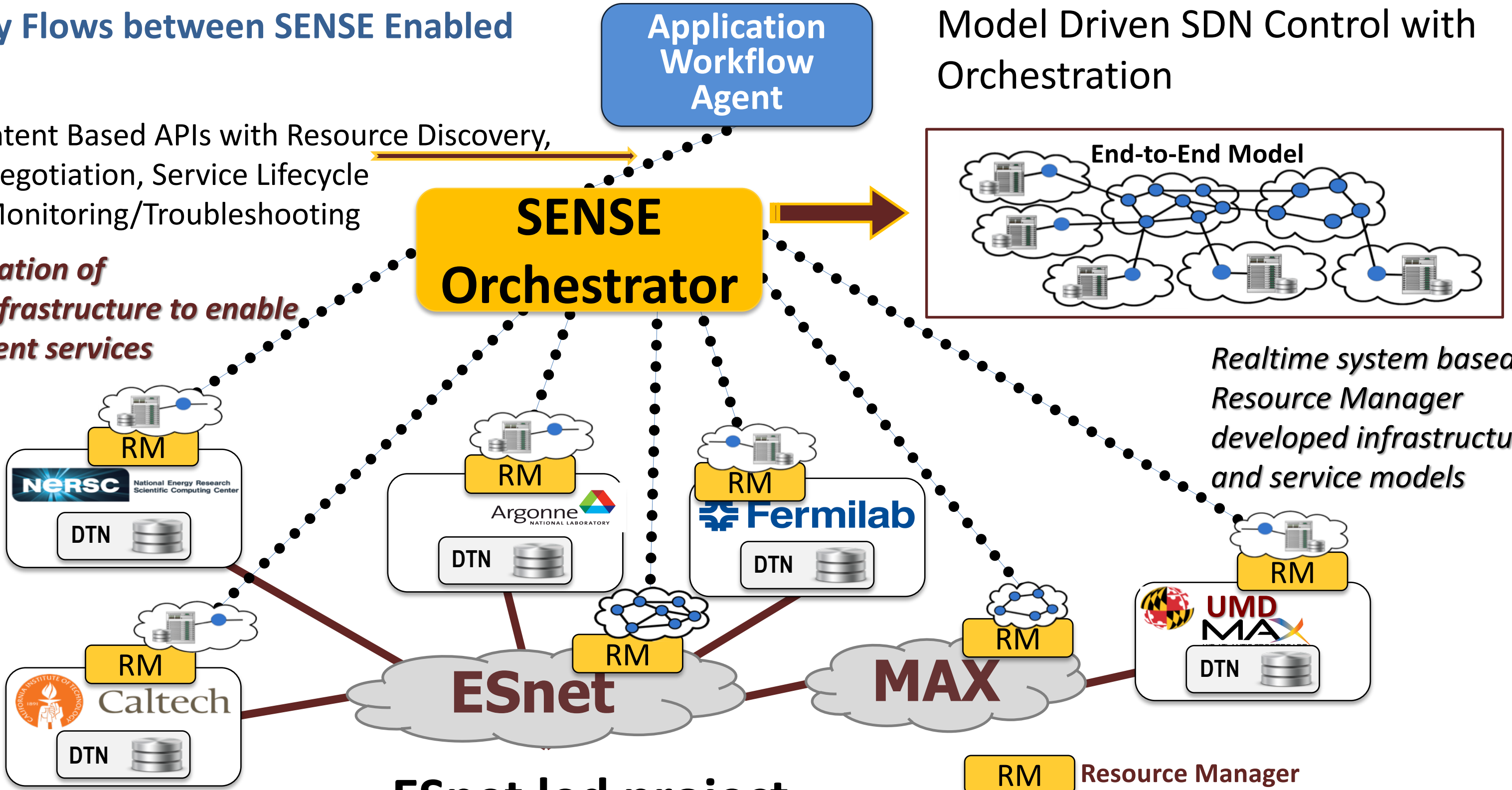
Priority Flows between SENSE Enabled DTNs

Model Driven SDN Control with Orchestration

Intent Based APIs with Resource Discovery, Negotiation, Service Lifecycle Monitoring/Troubleshooting

Datafication of cyberinfrastructure to enable intelligent services

Realtime system based on Resource Manager developed infrastructure and service models



RM Resource Manager

Generates model at resource owner level

ESnet led project

Demonstration Videos

- SD-SDMZ Hybrid Cloud Inter-Networking (HCIN) Advanced Hybrid Cloud (AHC) Service Demonstration Video:
  - https://tinyurl.com/max-sdmz-pilot
- EdgeCloud Connection Service Demonstration Video:
  - https://tinyurl.com/max-ecc-pilot

Thanks