SAVI Testbed
Initial Views

Hadi Bannazadeh, PhD

Testbed Architect
Smart Applications on Virtual Infrastructures
Electrical and Computer Engineering Department
University of Toronto
November 2011
Agenda

- Introduction
- VANI Architecture
- SAVI Testbed
  - Architecture
  - Features & Challenges
- Control and Management
- Resource Virtualization
- Resource Management
- SAVI Testbed Releases
- Conclusion
- Backup Slides (Resources Details)

11/10/2011
SAVI Testbed (TB)

- Playground for SAVI research themes

- Used and built by
  - SAVI Themes:
    - Applications Theme
    - Extended Cloud
    - Smart Edge
    - Wireless/Optical
    - Future Internet projects
    - Possibly by Researchers outside VANI

- Testbed Theme members:
  - **Leon-Garcia (lead), Boutaba, Chow, Ganjali, Li, Litoiu, Rusch, Steffan, and all other SAVI PIs**, 1 engineer, $1 + 4 \times \frac{1}{4}$ post-docs, 5 grads, 2 interns, 2 undergrads

11/10/2011
VANI to SAVI

- Virtualized Application Networking Infrastructure
  - A networking research testbed
    - Designed and prototyped at UofT
    - Based on UofT’s Application-Oriented Network architecture

- SAVI Testbed has roots in VANI
  - Lessons learnt from VANI
    - Architecture
    - Middleware
    - Development
    - ...

11/10/2011
VANI Architecture

Control and Management Plane

Abstraction Layer

Virtualization Agents

Web Service-Interface

Application Plane

Abstraction Layer

Virtualization Agents

Virtualized Resources

Physical Resources

WS

UDP/IP

SSH

HTTP/SSL

Control and Management (C&M)
VANI Architecture

Control and Management Plane

Application Plane

Physical Resources

11/10/2011
SAVI TB Architecture

SAVI TB Architecture

Control and Management Plane

Application Plane

Physical

Resources

11/10/2011
Control and Management

- Provides access to virtualized resources
- Handles Auth/Auth/Accounting (AAA)
- Holds SAVI configuration
- Facilitates measurement and monitoring
- Includes advanced services
  - PaaS
  - Autonomic Resource Management
  - ...

11/10/2011
SAVI TB Initial Features (VANI)

- Basic Control and Management Features
  - Slice Management and Resource Allocation
  - Configuration Management
  - Security Management (AAA)
- Processing Virtualization
  - Linux vServer
- Storage Virtualization
- Reprogrammable HW Resource
  - BEE2
- Network Services (IPv4)
  - Virtualization using Ethernet VLAN
  - GW Services (NAT, DNS, public-private mapping)
SAVI TB Main Features
(Beyond VANI)

- Advanced Security Management
  - Authentication/Authorization/Accounting, SingleSignOn, Trust Management, etc.
- Network Virtualization and Management
  - OpenFlow, IPv6, Raw Ethernet, Wider Area Ethernet
- Open-source cloud computing
  - e.g., openstack (computing, storage, image, network)
- Reprogrammable HW Virtualization (e.g., BEE3, NetFPGA)
- Platform as a Service
  - Adaptive Application and Resource Management
  - Edge/Cloud Application Deployment
  - Converged computing and networking resource management
- Measurement and Monitoring Services

11/10/2011
SAVI TB Main Design Decisions

- Connectivity
- Security

- Leveraging work in
  - Open-source cloud computing
  - Clean-slate and future Internet projects

- Providing advanced management services
  - Converged networking and computing resource management

- Federation

11/10/2011
Agenda

- Introduction
- VANI Architecture
- SAVI Testbed
  - Architecture
  - Features & Challenges
- Control and Management
- Resource Virtualization
- Resource Management
- SAVI Testbed Releases
- Conclusion
- Backup Slides (Resources Details)

11/10/2011
SAVI Testbed
Control & Management Plane

ESB/Middleware

Users

Control & Mgmt Abstraction Layer

Virtualization

Agents/Physical Resources

A A A

C&M

DB

Res A

Res A

Res A

Res B

Res B

Res B

UserA

UserB

UserC

11/10/2011
Main Functions:
- Provide abstraction of resource to SAVI Control and Management SW
- Follows the API template provided by SAVI-CM
- Follows a registration process with CM

Other features:
- Needs to establish trust with CM
- Could be distributed/redundant
- Sends notification on resources status/measurements to SAVI-CM,
  - According to a unified template
  - May be passed on to researchers
Virtualization Agents

- Abstraction layer’s arms in virtualizing a resource
  - Usually attached to a physical resource
  - Customized for each resource
  - Provides isolation between different slices of resource
  - Provides SLA to VANI-CM

- Uses SAVI middleware to communicate with its corresponding abstraction Layer

- May be able to directly send notification to:
  - Researcher/SAVI-CM/Accounting/Abstraction Layer
SAVI C&M APIs

- Different Classes of APIs
  - Programmable resources
    - Computing/HW/Sensor resources
  - Storage resources
  - Communication/Networking Resources
    - Wireless/Wired communications

- Initially follows VANI interface
A Sample Message Sequence

Researcher

Control

Virtualized Resource

Start

getResource

 getResource

Auth/Authz

Accounting/Record Keeping

programResource

Direct Connection to Resource
List of Resources (evolving)

- Essential resources:
  - Storage
  - Processing
  - Network (Fabric)
  - Gateway
  - Reprogrammable HW

- Secondary Resources
  - Advanced SAVI developed resources
    - Resource Management
  - 3rd party resources
3rd Party Resources

- Being able to incorporate independently developed resources
  - Advanced management services
  - Reusable service components
    - e.g.: Database, Sensors, Complex Event Processors, ...

- Challenges
  - APIs
    - Middleware and connectivity
    - Registration
  - Security and trust
  - SLAs
  - ...

11/10/2011
Autonomic Resource Management

- Adaptive Cloud Resource Management
  - Adapting resource consumption to the usage
  - Dynamic scaling of applications

- Network-constrained resource management
  - Optimum allocation of resources on the cloud and edge nodes
    - based on networking constraints

- Adaptive selection of cloud and edge resources

- Converged computing and networking resource management
  - Considering constraints and condition of computing, storage, networking, etc.
SAVI TB Releases

- Follows iterative approach
  - Delivers in releases
- Each release adds a subset of features in various areas
  - Iterative approach will be followed in delivering features as well
    - Multiple releases before fulfilling a feature
  - Minor releases: ?.x.y
  - Major releases: ?.0.0

- Goal:
  - to deliver desirable features by the end of each SAVI Milestone

11/10/2011
SAVI Releases

- SAVI 0.1 := VANI
- SAVI 0.2 += Min(Openstack-Storage)+Min(OpenFlow)+Min(Identity Manager)
- SAVI 0.3 += Nmin(Openstack-Storage) + Min(Openstack-compute,network) + Nmin(OpenFlow)+Min(Monitoring)
- SAVI 0.4 += Min(PaaS)+Nmin(IdMgr)+Nmin(Monitoring)
- SAVI 0.5 += Nmin(PaaS)+Min(ReprogrammableHW)
- ...

- SAVI 3.0 := DETS, OpenFlow, Openstack, Identity Manager and SSO, BEE3/NetFPGA,IPv6,PaaS

11/10/2011
Definition of Min/Nmin Features

- Next/Minimum features are defined at each releases for maximum three releases ahead
  - In collaboration with:
    - Members of testbed theme
    - Other themes leaders and post-docs and other partners
- Coordinated by testbed architect
Conclusion

- Initial views on SAVI TB architecture
  - Starting point, Goals, Challenges

- SAVI built and used by SAVI researchers (directly/indirectly)

- Main users/validator of SAVI TB:
  - Applications theme
  - Future Internet projects
  - Researchers outside SAVI
    - Attracting researchers outside SAVI; extremely important

- Users are strongly asked to use SAVI testbed releases
  - To help choose the right path in testbed design and development

11/10/2011