

On Bypassing Middle-Boxes Through Campuses

Petr Holub

<hopet@ics.muni.cz>

Laboratory of Advanced Networking Technologies

CESNET, Czech Republic

Masaryk University



2st E2E Workshop on Provisioning E2E Services

TERENA

Amsterdam, 2009–12–07



Talk Overview

Intro

CoUniverse

CoUniverse – NLM Usecase

VirtCloud

VirtCloud – Cluster Tunneling Usecase

Conclusions



Middle-Boxes and E2E Services

I'm looking for somebody to speak about the various middle boxes (firewall, proxy, NAT, etc.) and the related issues should be solved when using real e2e services.

... humpf...

... heck, aren't we building E2E services to avoid those thingies?

Why do we need E2E?

(revisited since 12/08)

- Get bandwidth to somewhere
 - ... workaround for flawed production networks
 - sometimes easier to solve last hop problem
 - sometimes easier to get funding for (it has to be research then)
- Traffic isolation and “guarantees”
 - important for applications with bitrates comparable to link capacities
 - to what extent can this be implemented in production networks?
- Cheaper L2 equipment
 - “lame” excuses of network administrators when it comes to 10GE ports on their routers ;-)
 - we’re down to 1.200 EUR per 10GE LR port w/o VAT



Why do we need E2E?

(revisited since 12/08)

- Low-jitter of L2 network
 - important, e.g., for HW devices with low buffering
- Some special properties
 - introduction of various devices on defined positions in the network
 - reflectors, optical multicasting switches, traffic monitors, etc.
- To avoid various middle-boxes
 - firewalls, NATs, proxies (maybe even monitoring & IDS systems that slow down network)
 - hard to get exceptions on a device that serves whole institution
 - some “paranoid” institutions run weird configurations (e.g., multi-layered proxies in one Czech hospital)
- To give user both functionality and *responsibility*
 - discussed later for VirtCloud

CoUniverse – Motivation

- High-definition collaborative environments
 - Using high-quality, high-definition media streams to build collaborative environment
 - ◆ bandwidth demands comparable to network link speeds (10 GbE) requires careful planing and configuration of infrastructure:
UltraGrid:
DXT-compressed HD video over IP: 250 Mbps
uncompressed 4:2:2 HD video over IP: 1.5 Gbps
uncompressed 4:2:2 4K video over IP: 6 Gbps
 - ◆ lacks adaptivity to changing networking conditions
 - Large numbers of components needed to build the environment
 - ◆ each one of them needs to be configured
 - ◆ hard to orchestrate them manually to build the desired environment
 - ◆ virtually impossible to cope with network events manually



CoUniverse – Motivation

CoUniverse & E2E Networks

- On-demand circuits require allocation
- Application-driven network allocation
 - if we can control applications, why not the network?
 - user should not be forced to allocate it manually
- CoUniverse is ideal for implementing this
 - it's just another component to orchestrate
 - framework is general enough to implement it



CoUniverse Building Blocks

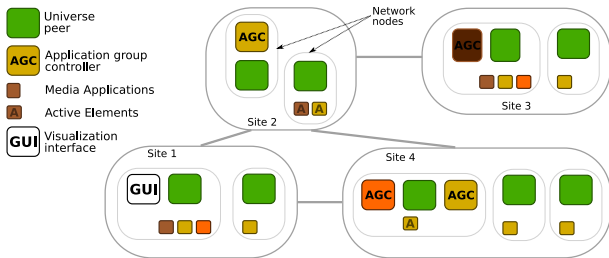
- Organization of the CoUniverse
 - Collaborative Universes where actual collaboration takes place
 - ◆ partitions virtual space (like, e.g., Virtual Venues)
 - ◆ provides privacy for users (may enforce authentication and authorization of users)
 - ◆ limits size of the system
 - Multiverse
 - ◆ registration and lookup of Collaborative Universes
 - ◆ automatically joined by each node
- Network organization
 - control plane based on P2P substrate
 - ◆ maximize robustness of the network
 - ◆ distributes control messages, updates from monitoring, etc.
 - one or more data planes running over native network
 - ◆ maximize performance for the applications (typically throughput, latency, jitter)



CoUniverse Building Blocks

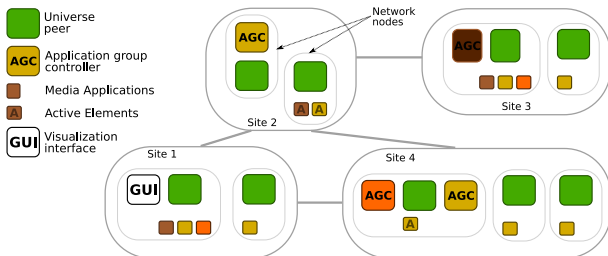
- Components

- nodes: physical nodes, having one or more network interfaces, running individual orchestrated applications
- proxy nodes: proxies and controllers for devices that can't run CoUniverse directly (Polycoms, microscopes, etc.)
- sites: aggregate of network nodes (e.g., you can tell you want stream from a specific site)
- application groups: aggregate of applications



CoUniverse Building Blocks

- Application Group Controller (AGC)
 - controls operations in the Universe
 - contains media streams scheduler if needed
 - one AGC per application group

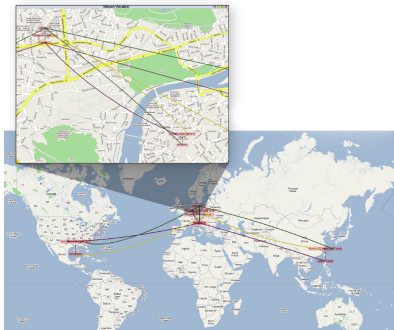


Self-organization in CoUniverse

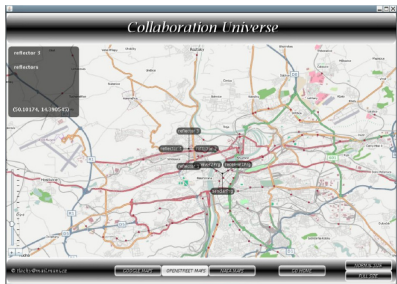
- Dynamic media streams scheduling:
 - Schedule media streams produced by media applications on particular network links (plan step)
 - scheduling media streams using bandwidth close to physical link capacity is hard
 - scheduling based on set of constraints
 - ◆ producer constraints, consumer constraints, data distribution constraints, network link constraints
- Resilience:
 - ability to react to changes/failures in the network infrastructure, media applications etc.
 - achieved by monitoring, infrastructure changes and/or failures lead to new media streams schedule

Visualisation in CoUniverse

- overview of actual CoUniverse state for the user
- network topology visualization
- actually scheduled media streams



GLIF 2007 visualisation



Current development visualisation

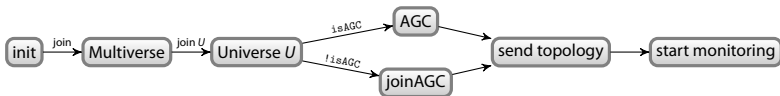


CoUniverse & Internet2 DCN

- DCN interfaces
 - web interface for humans
 - web service interface (w/ security!) for machines
- Initializing and tearing circuits
 - network link can have associated one or more lambda links
 - DCN-specific lambda link: two endpoints (including identification, IDC and (tagged|untagged) interface) and requested bandwidth
- Integration of on-demand circuits with CoUniverse brings another level of uncertainty into scheduling
 - should I count on a network link I'm uncertain to get?
 - should I preallocate the network? (but there are $\frac{n^2-n}{2}$ bidirectional links!)

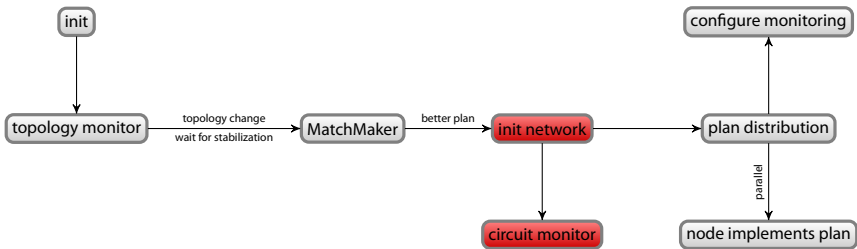


CoUniverse Node Operations Diagram



- after joining a universe, an existing AGC is either joined or new one is created
- sends topology to the AGC, including
 - end-to-end network topology
 - node configuration (site membership, interfaces, network connections, available applications)
 - requested sites for receiver applications
- waits for a new events from AGC (e.g., new plan) and monitors infrastructure to send updates to AGC

CoUniverse AGC Diagram



- MatchMaker

- finds suitable source based on configuration of each receiver (if possible)
- builds plan based on available network features (links, reflectors)

- network initialization

- added for end-to-end circuit initialization
- blocking stage to make sure we have the network prior to application startup

Notes on CoUniverse Implementation

- State of the CoUniverse
 - goal is to have usable proof-of-concept implementation
 - ◆ research on self-organization, application orchestration, scheduling
 - ◆ real-life applications (science, education)
 - <https://www.sitola.cz/CoUniverse>
- Open-source
- Implemented in Java, works on Linux, MacOS X, Windows
- P2P control plane
 - based on JXTA 2.4.1
- Internet2 DCN
 - OSCARS API using Axis and Rampart



CoUniverse – NLM Usecase

- We want to have videoconferencing with National Library of Medicine at National Institute of Health (NLM NIH), but...
 - there's not enough bandwidth to use UltraGrid (even DXT)
 - there's enough firewalls for H.323/SIP not to work
 - ... and we would like to have both
 - and for pathology, we'd even love to do SAGE-based tiled screen

CoUniverse – NLM Usecase

- Washington, DC – Bethesda, MD
 - collaboration with National Library of Medicine
- Washington, DC – Houston, TX
 - collaboration with Memorial Hermann Texas Medical Center
- What do we need to make this happen?
 - network initialization
 - monitor network and wait until it comes up
 - startup of all applications and components
 - monitor everything and react to events
 - termination of all applications and components
 - network tear-down
- ... we don't want this by hand, do we?



Network Topology for NLM Usecase

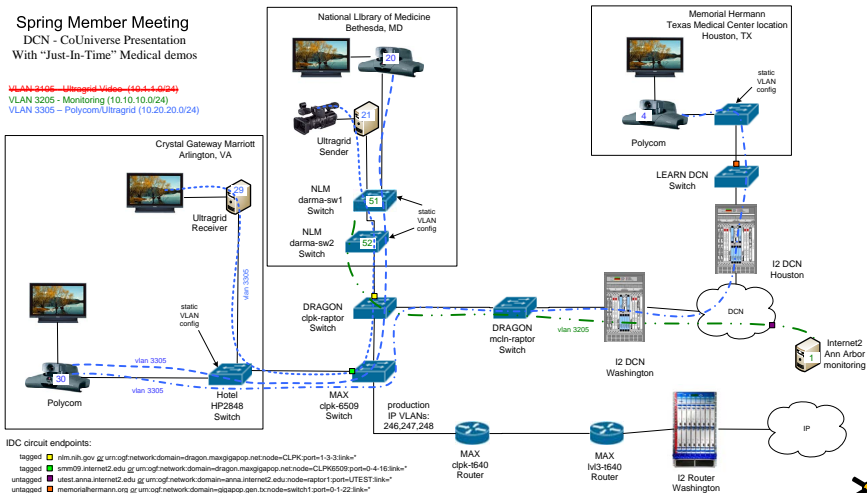
- CoUniverse can
 - orchestrate not only applications, but also on-demand network circuits
 - orchestrate several types of H.323 Polycom devices
- Texas has decent Internet2 DCN connection after several demo events
- Bringing DCN to NLM showed much easier than
 - on-demand network is strictly separated from normal NLM traffic
 - avoids problems with breaching security of network carrying patients' data
 - has sufficient performance



CoUniverse – NLM Usecase - Topology

Spring Member Meeting DCN - CoUniverse Presentation With "Just-In-Time" Medical demos

VLAN 2405 – Ultragrid Video (10.1.4.0/24)
VLAN 3205 – Monitoring (10.10.10.0/24)
VLAN 3305 – Polycorn/Ultragrid (10.20.20.0/24)



IDC circuit endpoints:

tagged ■ nlm.nih.gov `gr um:ogf:network:domain=dragon.maxgigapop.net:node=CLPK:port=1-3-3link="`
 tagged ■ smm09.internet2.edu `gr um:ogf:network:domain=dragon.maxgigapop.net:node=CLPK6509:port=0-4-16link="`
 untagged ■ utest.anna.internet2.edu `gr um:ogf:network:domain=anna.internet2.edu:node=raptor1:port=UTESTlink="`
 untagged ■ memorialhermann.org `gr um:ogf:network:domain=gigapop.gen.bcnode=switch1:port=0-1-22link="`

MAX/DRAGON IDC URL: <https://idc.dragon.maxgigapop.net/taxis2/services/OSCARs>

Wed Apr 22 2009



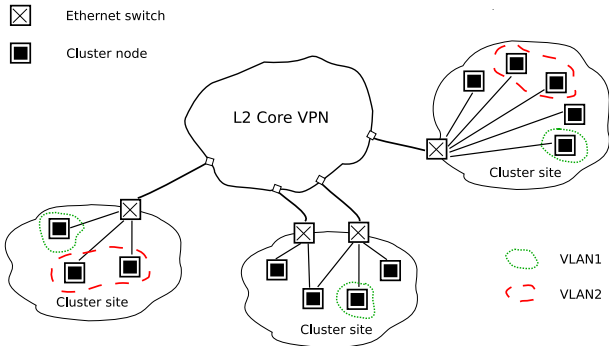
VirtCloud

- Virtualization changes the way Grids can work
 - interactive jobs based on preemption
 - migration of virtual (typically computational) nodes
 - nodes may run insecure (user-provided) virtual machines
- We need networks that can support these usecases
 - reasonable setup time
 - private network by default
 - migration of IP addresses (which need to be retained by the applications in order to ensure uninterrupted operation)
 - high performance, low latency, low jitter needed (data transfers, MPI communication, etc.) \implies minimize the penalty



VirtCloud

- Virtual network for virtual clusters
- Managed by the virtual cluster management system
- Architecture



- flat private L2 network for each virtual cluster

VirtCloud

- Architecture (cont'd)
 - core network
 - ◆ configured only once to carry the VLANs needed
 - ◆ should use some native technology (QinQ, VPLS, Xponders)
 - site network
 - ◆ this is dynamically reconfigured to enable VLANs for individual physical hosts
 - physical node
 - ◆ hosts one or more virtual nodes
 - ◆ each virtual node can access one VLAN
 - ◆ physical host uses trunking to connect to the site network
 - incorporation of additional services if needed
 - ◆ service virtual machine: runs, e.g., DHCP server, (proxy) NFSv4 server
 - possible tunneling to user's site

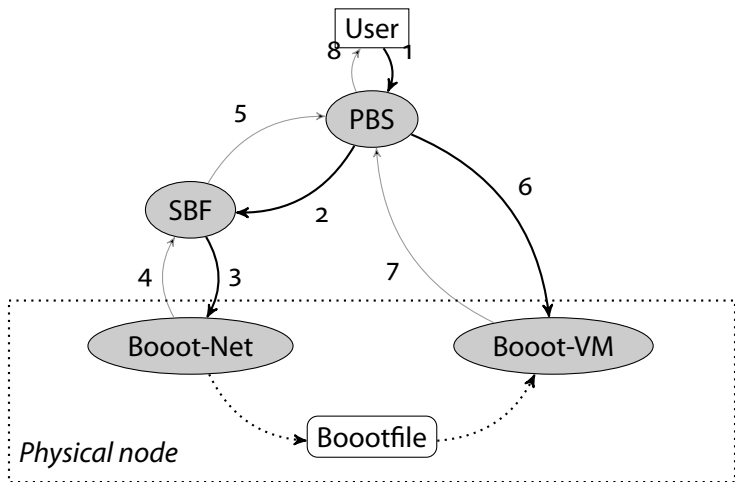


SBF: Slartibartfast Prototype

- Simple implementation, that virtual cluster management system Magrathea can interact with
- Uses Ethernet over CESNET2 network
- Core network: VPLS or Xponders
 - VPLS uses production MPLS backbone (shares traffic)
 - Xponders run over a separate λ over production DWDM system
- Site networks: Brno, Prague, Pilsen
 - L2 equipment from multiple vendors: HP, Force10, Cisco



SBF: Slartibartfast Prototype—Interactions



VirtCloud – Cluster Tunneling Usecase

- Giving user access to the cluster
 - to have impression it's *my* cluster
 - there is often many various middle-boxes between the cluster and the user
- Defaults to simple SSH to cluster headnode
- Gives an OpenVPN option
 - service node running service image that provides OpenVPN server with public address
 - bridged mode gives user impression that his station is part of the cluster
 - OpenVPN works surprisingly well
 - OpenVPN is easy to get not even across firewalls & NATs, but also most HTTP proxies

	no VPN	UDP VPN	TCP VPN	TCP VPN + HTTP proxy
pchar latency [ms]	3.51	3.69	3.94	3.93
iperf jitter [μ s]	6	6	9	13
pchar capacity est. [Mb/s]	39.8	35.2	20.1	19.8
iperf packet loss @ 30 Mb/s [%]	0.0	0.0	0.0	0.0
iperf CPU idle @ 30 Mb/s [%]	48.9±0.2	41.7±0.4	44.5±0.4	42.6±0.4



VirtCloud – Cluster Tunneling Usecase

- Accessing user's resources from within the cluster
 - access to data from user's site
 - access from within the cluster to Internet is limited for security reasons
- OpenVPN-based approach
 - using OpenVPN allows incorporation of most resources from user's site
 - bridging on the OpenVPN client allows to interconnect whole networks
 - multiple connections to OpenVPN server: certificate-based authentication
 - ◆ getting multiple machines without whole network: e.g., NFS server,
- When user's site is available through on-demand service, it can be reached directly



VirtCloud – Cluster Tunneling Usecase

- Publishing cluster within user's own responsibility realm
 - virtual cluster can run arbitrary user-provided images
 - no access is provided to public internet
 - ◆ limited on Layer 2
 - user can use OpenVPN tunnel to publish the cluster under own address space
 - ◆ NAT—cluster behind single user's address
 - ◆ optional customized DHCP server for the cluster to provide addresses from user's address range
 - ◆ user has sole responsibility
 - ◆ segregated on Layer 2 from other clusters
- Only certified images are allowed to access internet directly



Conclusions

- Firewalls, NATs, proxies – annoying for advanced applications but probably unavoidable
- It makes sense to build E2E services for various reasons
 - ... and one of them is to bandwidth and open network to confined applications in security-sensitive institutions
- Tunneling to bridge the gap between the networks
 - also works reasonably well for firewalled networks
- Worthwhile to participate in things like OGF NSI-WG for developers and advanced users
 - <http://forge.gridforum.org/sf/projects/nsi-wg>

References

- (1) LIŠKA, Miloš – HOLUB, Petr. CoUniverse: Framework for Building Self-organizing Collaborative Environments Using Extreme-Bandwidth Media Applications. In Lecture Notes in Computer Science vol. 5415 Euro-Par 2008 Workshops - Parallel Processing. Las Palmas de Gran Canaria, Spain : Springer Berlin / Heidelberg, 2008. ISBN 978-3-642-00954-9, pp. 339-351. 2008, Las Palmas de Gran Canaria, Spain.
- (2) ANTOŠ, David – MATYSKA, Luděk – HOLUB, Petr – SITERA, Jiří. VirtCloud: Virtualising Network for Grid Environments–First Experiences. In The 23rd IEEE International Conference on Advanced Information Networking and Applications AINA 2009. Bradford, UK : IEEE Comp. Soc., 2009. ISBN 978-0-7695-3638-5, pp. 876-883. 26.5.2009, Bradford, UK.



Thank you for your attention!

Q?/A!

<hopet@ics.muni.cz>

