



# Autonomic Adaptation of Virtual Distributed Environments in a Multi-Domain Infrastructure

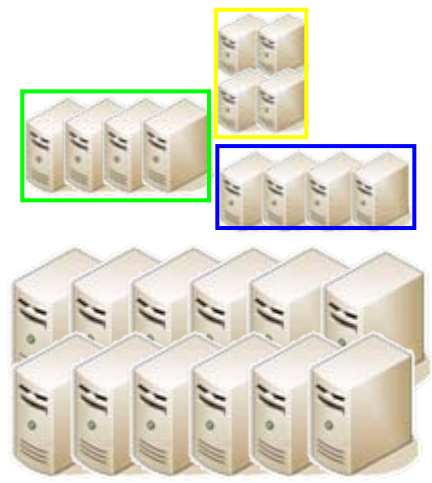
Dongyan Xu, Paul Ruth, Junghwan Rhee  
*Department of Computer Science*  
*Purdue University*

Rick Kennel, Sebastien Goasguen  
*Rosen Center for Advanced Computing*  
*Purdue University*

Presented By:  
Ryan Riley  
*Department of Computer Science*  
*Purdue University*



- Formation of *shared distributed cyberinfrastructure* (CI)
  - Spanning multiple domains
  - Serving users/user communities with diverse computation needs
  - Exhibiting dynamic resource availability and workload
- Need for *virtual distributed environments* (VIOLINs), each with
  - Virtual machines (VMs) on a virtual network (VN)
  - Customizability and legacy application compatibility
  - Administrative privileges
  - Isolation, security, and accountability
  - **Autonomic adaptation capability**
    - ✓ Adapt the environment to the needs of the application on the fly
    - ✓ A unique opportunity brought by virtualization (VMs and VNs)



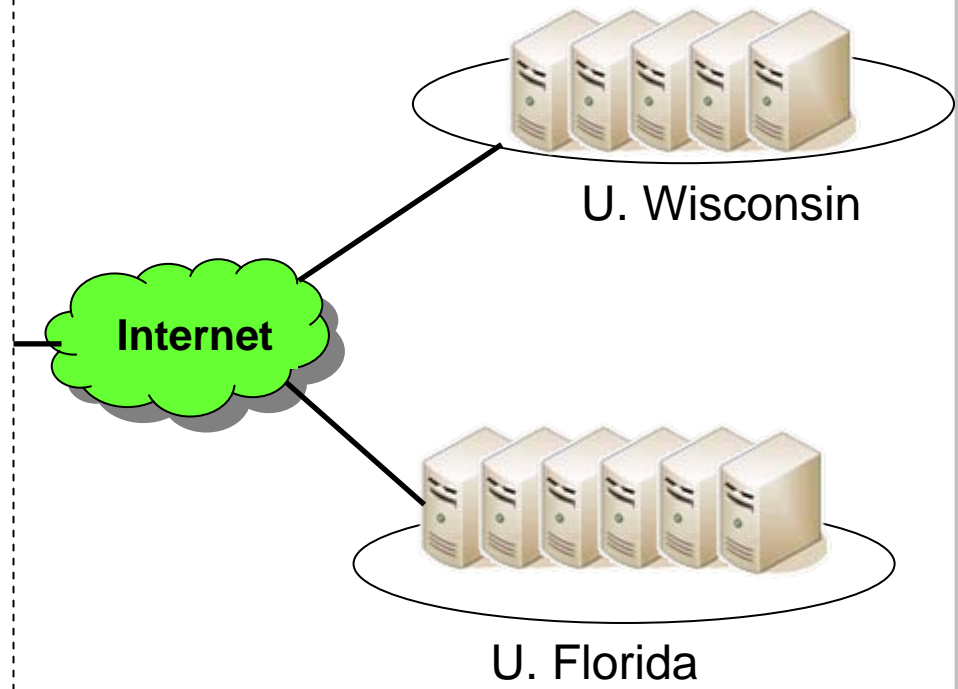
Virtual clusters  
(VIOLINs)



Physical cluster



nanoHUB infrastructure@Purdue





- *How can we adapt?*
- Requirements for multi-domain live adaptation mechanisms
  - Application transparency
  - Runtime resource scaling (CPU, memory, and network bandwidth)
  - Live virtual machine migration
    - ✓ Virtual machines retain same IP address
    - ✓ Cannot rely on consistent access root file systems (i.e. through NFS)
- This is mostly complete
  - Live migration within LAN (Xen, VMware)
  - Resource scaling in XEN
  - Live migration over WAN (VIOLIN)



- How can we adapt *quickly*?
- One thought is to use *depots*
  - VM system images are broken into parts (libraries, packages, COW)
  - Parts are strategically distributed throughout the network for easy access
  - The hope is that different VM images will share many parts
- Open Questions
  - Given a set of depots with various VM parts and a list of parts that compose a specific VM, find a fast way to construct the VM at a given host.
  - Existing content distribution networks may be useful to assist this effort.



- *When* should we adapt?
- Properties we can measure and change
  - Automatic VM resource monitoring and scaling
  - Application profiling and non-intrusive sensing of application needs
- Adaptation policy questions
  - How do we sense when adaptation will help an application?
  - What should we adapt?
    - ✓ Simple resource scaling or full fledged migration?
    - ✓ What resources should we try to increase? (CPU, memory, bandwidth, etc.)
  - Where should we migrate to?
  - How do you prioritize various VIOLINs?



- When should we *not* adapt?
- Security of hosted VIOLINs
  - Detection of compromise
    - ✓ Malicious code
  - Detection of abuse
    - ✓ Selfish users
    - ✓ Malfunctioning code
- How do we know a VIOLIN's resource demands are legitimate?
  - Need for justification and approval of adaptation requests
  - Approval should come from outside the VMs
- What actions should be taken to correct misbehavior?



A screenshot of a web browser window displaying the nanoHUB.org interface. The browser address bar shows a URL starting with 'http://www.nanohub.org...'. The main content area is divided into several panels. At the top, there are two side-by-side tables with columns for 'Name', 'CPU', 'Memory', and 'Disk'. Below these are two more similar tables. At the bottom, there is a terminal window displaying system boot logs, including lines like 'vmlinux (CPU) on (0,0,0,0)', 'vmlinux (GPU) on (0,0,0,0)', and 'vmlinux (GPU) on (0,0,0,0)'. The browser's taskbar at the bottom shows several open tabs, including 'nanoHUB.org - Home'.



# Video



- Condor (U. Wisconsin)
- VNET (Northwestern U.)
- Cluster-on-Demand (COD) (Duke U.)
- Virtual Workspaces on Grid (Argonne National Lab)
- SoftUDC (HP Labs)
- WOW and IPOP (U. Florida)



- Autonomic adaptation is an exciting research topic in virtual distributed environments
- There are plenty of open research questions
- Current results are promising
- Consider joining us in researching this area!



# Thank you.

Comments and collaborations are welcome!

Email: [dxu@cs.purdue.edu](mailto:dxu@cs.purdue.edu)

URL: <http://www.cs.purdue.edu/~dxu>

Google: "Purdue VIOLIN FRIENDS"

[http://www.cs.purdue.edu/~dxu/HPDC06\\_demo.mov](http://www.cs.purdue.edu/~dxu/HPDC06_demo.mov)