An Introduction to the Grid Appliance

Grid Appliance Development Team
ACIS Laboratory
http://wow.acis.ufl.edu
Summary

- Target audience
  - First-time users of the Grid appliance

- Material covered
  - Basic capabilities
  - Usage examples
  - Enabling technologies
  - Related systems
  - Quick start guide
Target users

- Our target user could benefit from Grid computing but doesn’t know how to start
  - I have simulation programs which take long to execute
    - And I have several parameters to simulate
    - Running on a single desktop takes too long
  - I want to exploit the computational capability of a “Grid”
    - But I don’t want to change your programs
    - I want a plug-and-play solution that runs on my desktop
Grid appliance in a nutshell

- Easy-to-install software with a pre-configured Grid environment
  - “Virtual machine” appliance; open-source software running on Linux, Windows, Mac
- Hands-on examples, bootstrap Grid infrastructure, and zero-configuration software – you’re off to a quick start
  - 15-30 minutes from download to job submission
Grid appliance in a nutshell

- Quick start and examples – you become familiar with capabilities and limitations
- You can continue to use the bootstrap Grid infrastructure
  - Or create a Grid of your own
- Creating an equivalent Grid on your own network is also very easy
  - Copy the same appliance to clusters, PC labs
  - Simple management – easy on your system administrator, even you can manage a small Grid
Use-case example (1)

Ann is a researcher; she sometimes needs computing cycles for several long-running simulation experiments, but can make cycles from her computer available to others in her research community when idle.

She downloads and runs a Grid appliance, creating a virtual machine (VM) in her own desktop that is now part of the appliance community.
Use-case example (2)

Bob, Carol, Diego, Esther, … are other users belonging to the same community and with similar needs; they also download and run the appliance
Use-case example (3)

Although they are distributed and on different computers, all the appliances are configured in the same way, and are connected by a virtual network, so applications that Ann tests on her own appliance can run on other appliances.

Now Ann can submit simulation jobs to the pool of community resources; they will make use of idle computing cycles from other users.
Use-case example (4)

When Ann’s appliance is idle, it is made available for others in the community to execute their own jobs.

Because jobs run in a separate, virtual environment, Ann’s computer is isolated from unintended or malicious behavior arising from programs that run within the appliance.
One appliance, multiple uses

The same appliance software:

• Works in an open wide-area environment
  • E.g. our global bootstrap infrastructure
• Works in a local-area environment
  • E.g. your department, lab, college
• Works in a private wide-area environment
  • A “virtual organization”
  • Strong public key infrastructure for security; can connect resources across a restricted set of institutions
What’s inside the Grid appliance

- It is a “virtual machine” with its own operating system and software
  - Linux and Grid middleware
- It connects to other appliances in a “virtual network”
  - Like a VPN, but P2P (U. Florida’s IPOP)
- It runs a robust Grid middleware scheduler to manage user jobs
  - U. Wisconsin’s Condor
What do you need to run it?

- Intel/AMD PC, laptop, server running Linux, Windows or MacOS
- Virtual machine (VM) software
  - Free: VMware Player/Server
  - Open-source: VirtualBox, KVM
- Your appliance runs like any other program in your computer
  - You don’t need to go through lengthy process of configuring your computer to be a Grid node; all you need is to install the VM
Why does it use VMs?

- Multiple operating systems run simultaneously and isolated from each other on the same computer
  - E.g. Windows “host” and Linux “guest”
  - Sandbox: bugs/crashes are confined to the VM
  - You can power off the VM if needed
- Support existing, unmodified applications
- Near-native performance for computationally-intensive tasks
Why does it use Condor?

- High-Throughput Computing (HTC)
  - Goal: effective management and exploitation of available resources

- Key technology: Condor
  - Developed at University of Wisconsin; part of the NSF Middleware Initiative (NMI)

- Key features:
  - Distributed ownership of resources
  - Harvesting of idle compute cycles
  - Robustness
How is it related to other systems?

- SETI@Home, Folding@Home, and other “voluntary computing” efforts
  - The Grid appliance allows its users to both contribute and use resources
  - The Grid appliance is not specific to an application; it can run your existing programs
  - You can deploy your own Grid pools on resources at your institution (or with your collaborators across the Internet)
Quick start

- Go to [http://wow.acis.ufl.edu](http://wow.acis.ufl.edu)
- Follow the “quick start” instructions:
  - Download a VM (e.g. VMware Player)
  - Download the appliance image (generic or configured to a specific project, e.g. CI-TEAM)
  - Turn the appliance on and connect to our open Grid
  - Go through a simple tutorial
- Give us feedback through our survey!
For more information

- This presentation serves as a short, basic introduction to the Grid Appliance
- Additional documents, papers, presentations, videos are available at:
  - http://wow.acis.ufl.edu
Credits and acknowledgments

- Development team at the ACIS Lab, U. Florida
  - [http://ipop.acis.ufl.edu](http://ipop.acis.ufl.edu)
- CI-TEAM at U. Florida
  - [http://ci-team.acis.ufl.edu](http://ci-team.acis.ufl.edu)
- Condor team at U. Wisconsin
  - [http://www.cs.wisc.edu](http://www.cs.wisc.edu)
Sponsors

- National Science Foundation
  - Under grants SCI-0537455, SCI-0438246, ANI-0300118

Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.