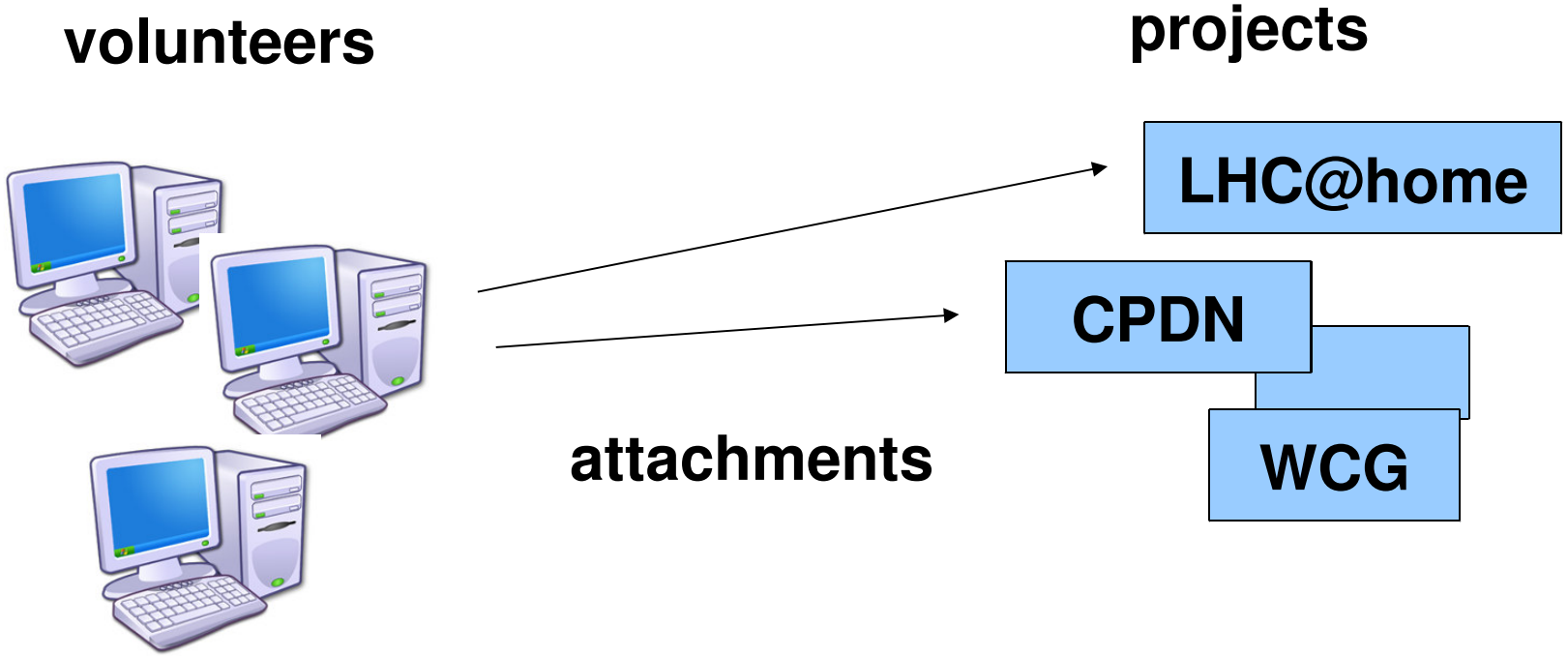
The Limits of Volunteer Computing

Dr. David P. Anderson
University of California, Berkeley

March 20, 2011

Volunteer computing



- Scientists create projects using BOINC
- Volunteers install BOINC, attach to project(s)
- Applications are silently downloaded and executed on volunteer PCs

To what extent can volunteer computing supplement or replace clusters, clouds, and supercomputers for HPC?

Volunteer computing status

- 40 projects
- 500K volunteers
- 800K computers
  - 2.4 cores/computer
  - 65% average availability
- 14 PetaFLOPS
  - would cost $5 billion/year on Amazon EC2

Potential

- 1.5 billion PCs
  - Graphics Processing Units: TeraFLOPS
  - Terabyte-scale storage
  - Network speed approaching 1 Gbps
- ~1000 ExaFLOPS today

The yearly cost of 10 TeraFLOPS

- Amazon EC2
  - small instance: $.09/hour = $788/year
  - 10 TeraFLOPS = 5,000 instances
  - $3.94M/year plus network, storage costs
- Build your own cluster
  - $1.5M/year
- Volunteer computing
  - $0.1M/year

Large-RAM jobs

- Hosts:
  - 31% have 4+ GB RAM
  - 9% have 8+ GB
  - 1% have 16+ GB
- User preferences on RAM usage
- Client monitors working-set size
- Scheduler: preferentially send large-RAM jobs to large-RAM clients

Long (week, month) jobs

- Checkpoint/restart features
- Preemptive scheduling
- Trickle up/down messages
  - communicate with server mid-job
- Intermediate file upload

Storage

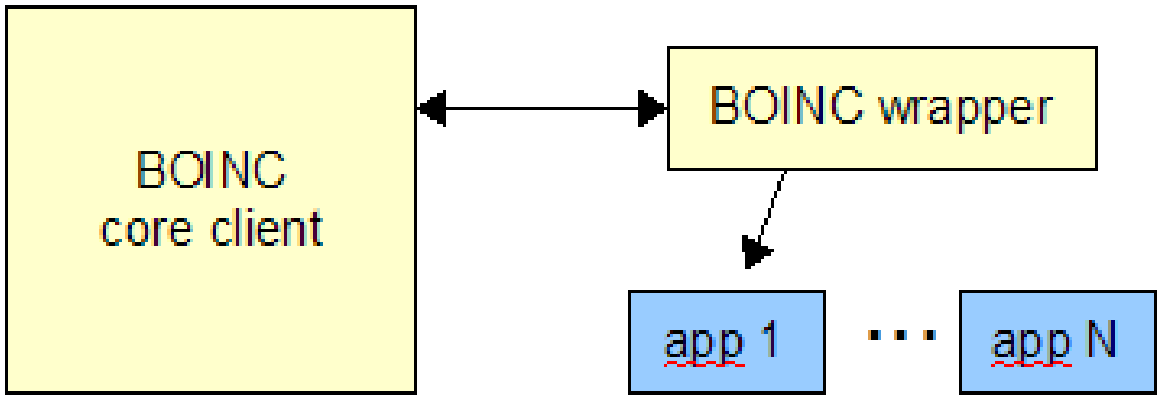
- Hosts:
  - 87% have 10+ GB free
  - 43% have 100+ GB free
  - 7% have 500+ GB free
- User preferences on disk usage

Communication

- Network bandwidth: hard to measure, but typically > 1 Mbps and growing
- User preferences
  - idleness, time of day, BW limit, monthly transfer limit
- BOINC file transfers
  - parallel
  - concurrent with computing
  - restartable

Support for diverse apps

- Native BOINC app
  - Link your app with BOINC library
- The BOINC wrapper
  - use with any existing app



Parallel apps

- Hosts
  - 27.6% have >= 4 CPUs
  - 7.1% have >= 8 cores
  - 0.4% have >= 16 cores
- BOINC client is multi-thread aware
- MPI wrapper
  - multi-host MPI: possible but hard
- Scheduler: preferentially send multicore jobs to multicore hosts

GPU apps

- Hosts
  - 27% have usable NVIDIA GPU
  - 14% have usable ATI GPU
  - 0.1% have both
- BOINC client recognizes, schedules GPUs
- User preferences for GPU usage
- Can use CUDA, CAL, OpenCL

Apps that run in VMs

- Reduces heterogeneity issues
- Sandbox for untrusted apps
- An app consists of
  - VM image, application, input files
- BOINC VM wrapper
  - instantiates VM (VirtualBox)
  - maps application, input files into VM
  - starts/suspends/resumes VM

Low-latency jobs

- Knowledge of host availability
- Priority-based scheduling in client and server

Non-technical issues

- Need to think big!
  - Single-scientist projects are a dead end
    - Barriers to entry are too high
    - Wrong marketing model
  - Umbrella projects
    - e.g., campus-level (UCBerkeley@home)
  - Concentrate on the public
    - don’t bother with in-house prototypes
- Resistance from HPC community

Conclusion

- Volunteer computing offers more/cheaper resources than other paradigms
- BOINC supports many types of HPC jobs
  - parallel
  - large resources requirements
  - VM-based
- Non-technical barriers remain