Wrocław University of Technology

Virtualization - What Can We Learn from Commercial Datacenters?

Igor Wawrzyniak
Wroclaw Centre for Networking and Supercomputing

Outline

- Intro to virtualization
- Virtualization and grids
- Pros and cons
- VM management
- Middleware
- Current status and plans

Virtualization

"All problems in computer science can be solved by adding another layer of indirection"

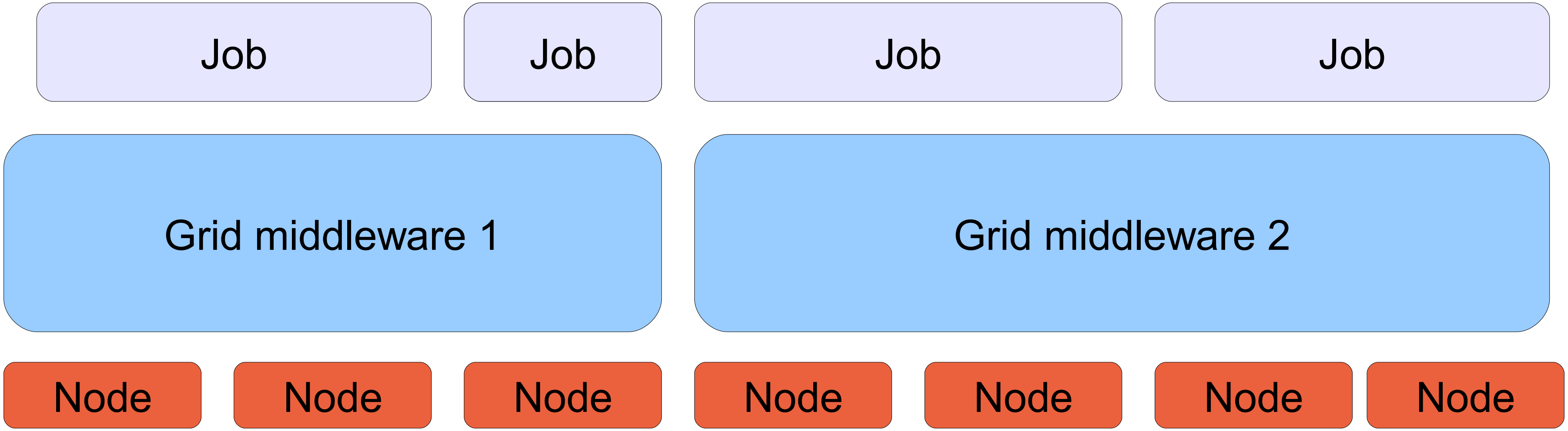
David Wheeler

"...except for the problem of too many layers of indirection"

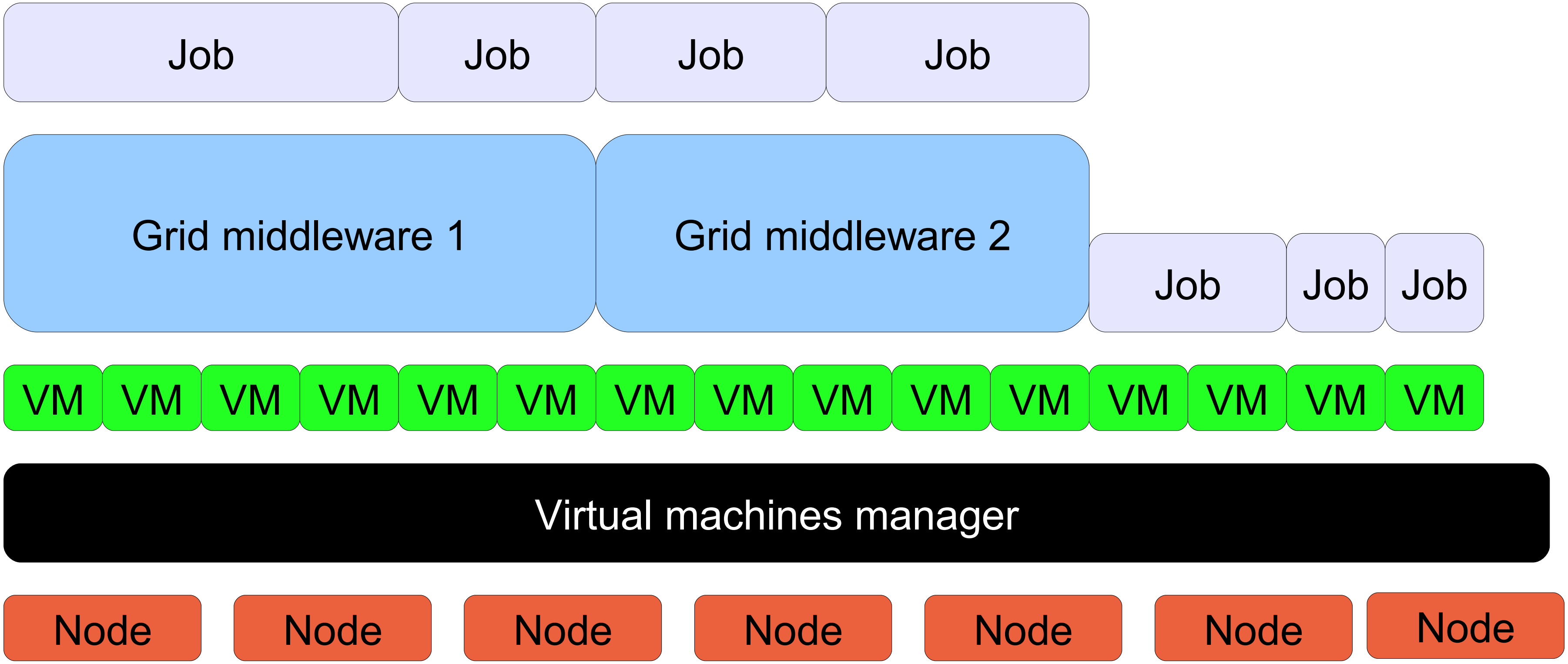
Kevlin Henney

Traditional grid architecture

- incomptabile standards
- static architecture
- only grid-aware software



Virtualized grid architecture



Job Job Job Job

Grid middleware 1

Grid middleware 2

Job Job Job

VM VM VM VM VM VM VM VM VM VM VM VM VM VM VM

Virtual machines manager

Node Node Node Node Node Node Node

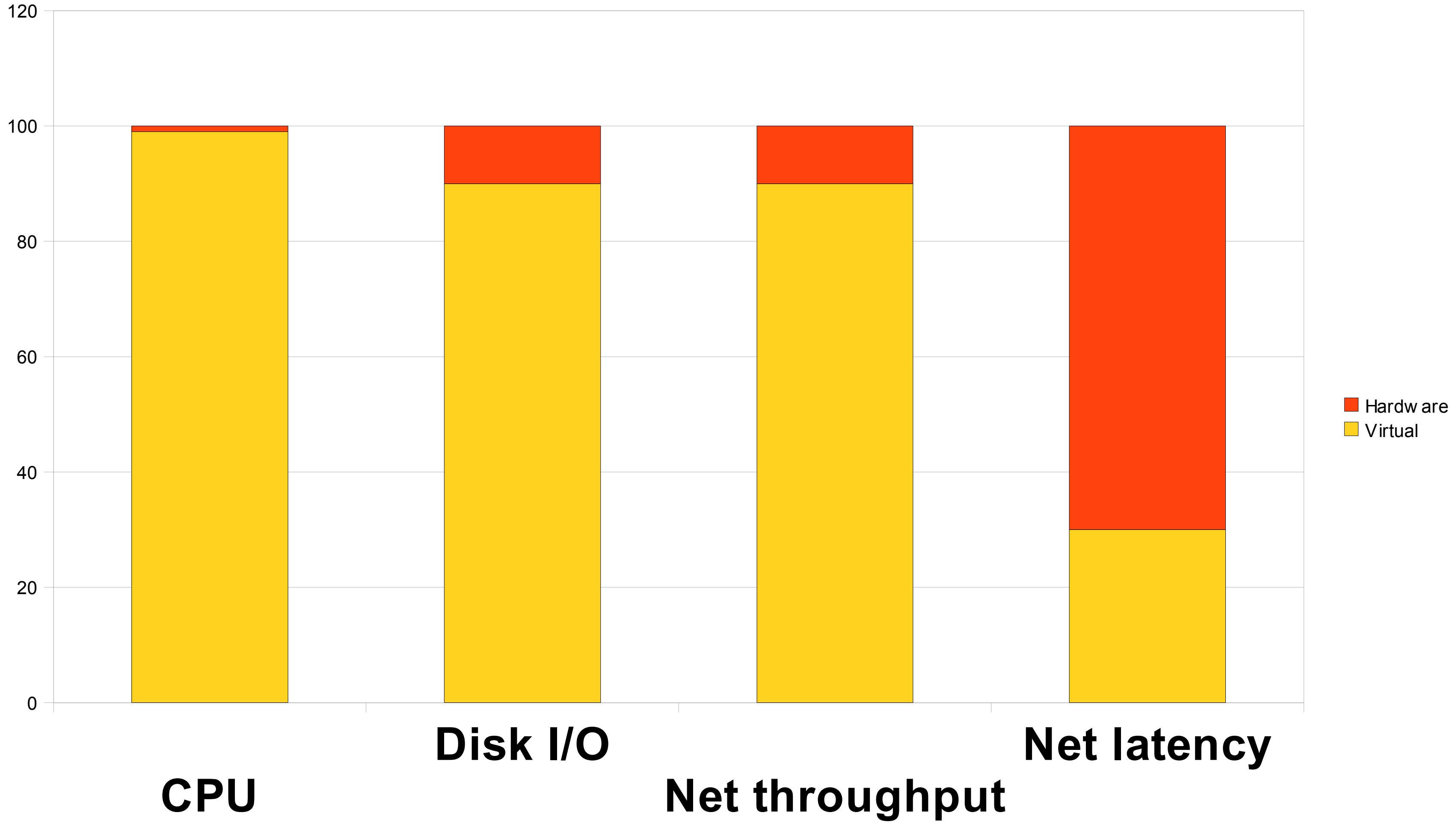
Pros

- Dynamic architecture
- Easy upgrade and testing
- Isolation and privacy
- Full user control of execution stack
- Able to run unmodified software
- Easier to maintain?

Cons

- Performance overhead
- Security issue – untrained operators with full OS control
- Less accurate timing
- Harder to maintain?

Virtualization overhead



Cloud platforms

- Amazon EC2
- Eucalyptus
- Nimbus
- VGRMS

Eucalyptus

- University project turned commercial
- Users: Ubuntu Cloud, NASA, Eli Lilly
- Open source
- Based on EC2 architecture, compatible API
- VMs controlled by user

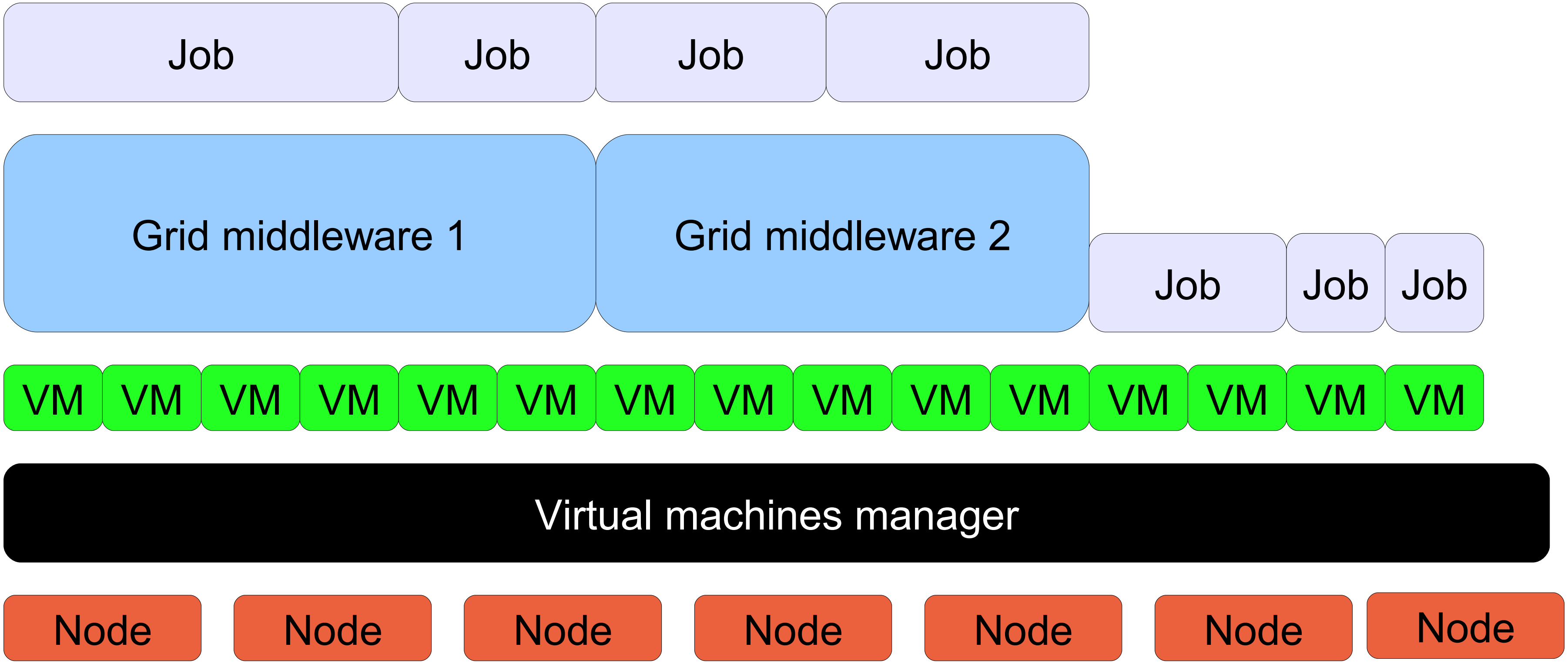
Virtual Grid Resource Management System

- 2-layer mapping: hardware to VM, VM to application; LDAP based
- Centralized provisioning of physical machines (DHCP, TFTP)
- User access: either direct or through grid portal
- Java-based administrator GUI for management and monitoring – also mobile version

VGRMS - storage subsystem

- Persistent storage
- Based on iSCSI and ZFS
- Clone, snapshot, restore
- Copy-on-write

Virtualized grid architecture again



Job Job Job Job

Grid middleware 1

Grid middleware 2

Job Job Job

VM VM VM VM VM VM VM VM VM VM VM VM VM VM VM

Virtual machines manager

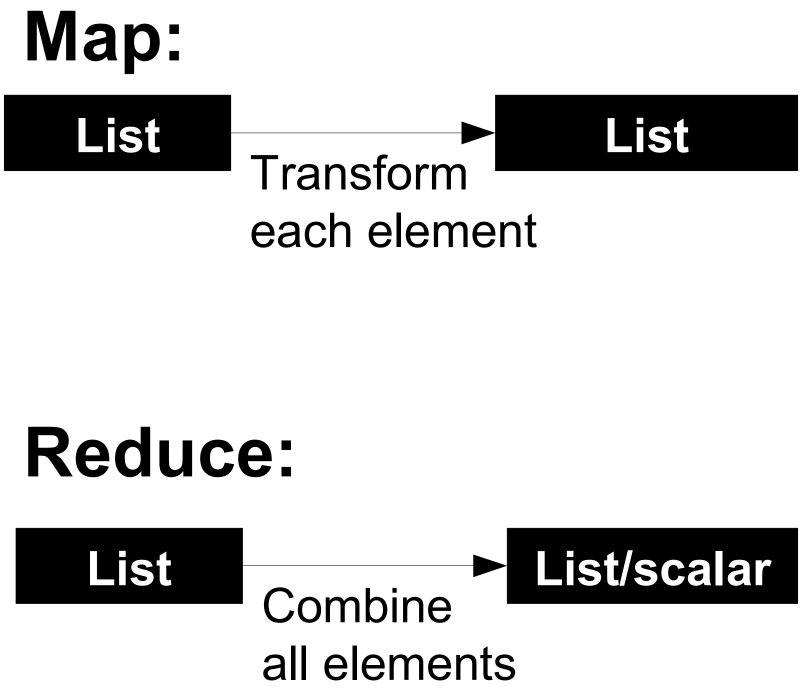
Node Node Node Node Node Node Node

Middleware

- Bare VMs:
  + flexible
  + any software
  - cooperation is complex

- Grid
  + more flexible than traditional grid
  - only if your software is grid ready

MapReduce, Hadoop



Map:

List → List

Transform each element

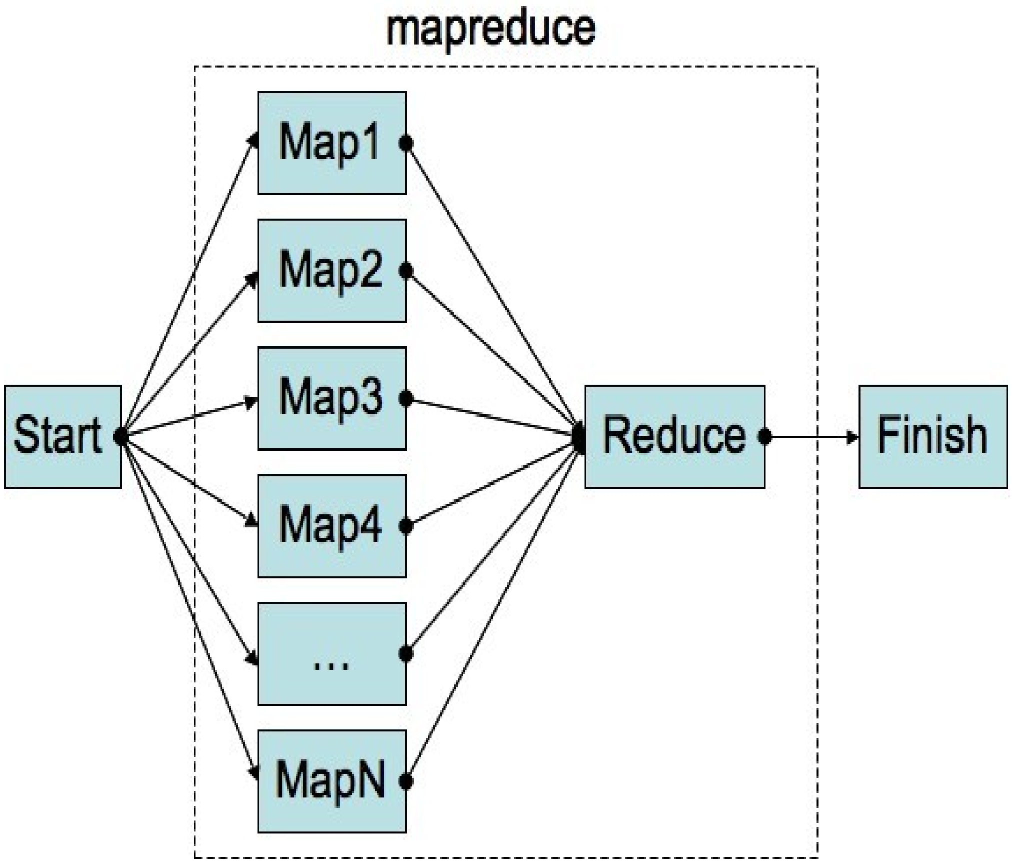
Reduce:

List → List/scalar

Combine all elements

- Standard example: word count
- Usage: data indexing (Google!) and mining, genetics, biochemistry, language processing

MapReduce, Hadoop



- Framework does distributing, collecting, failure discovery
- Designed for slow networks and unreliable components

Our experiments

- First experiments with Eucalyptus
- To be replaced by VGRMS
- VMs spawned on demand
- On top of that: gLite, Unicore, Hadoop
- ...or direct access to VMs