



CompSci 401: Cloud Computing

Automation Contexts

Prof. Ítalo Cunha



Groups that use automation

- Individual customers
- Start-ups and enterprise customers
 - Individual developers
 - Large companies using public clouds
- Cloud providers

Automation for end users

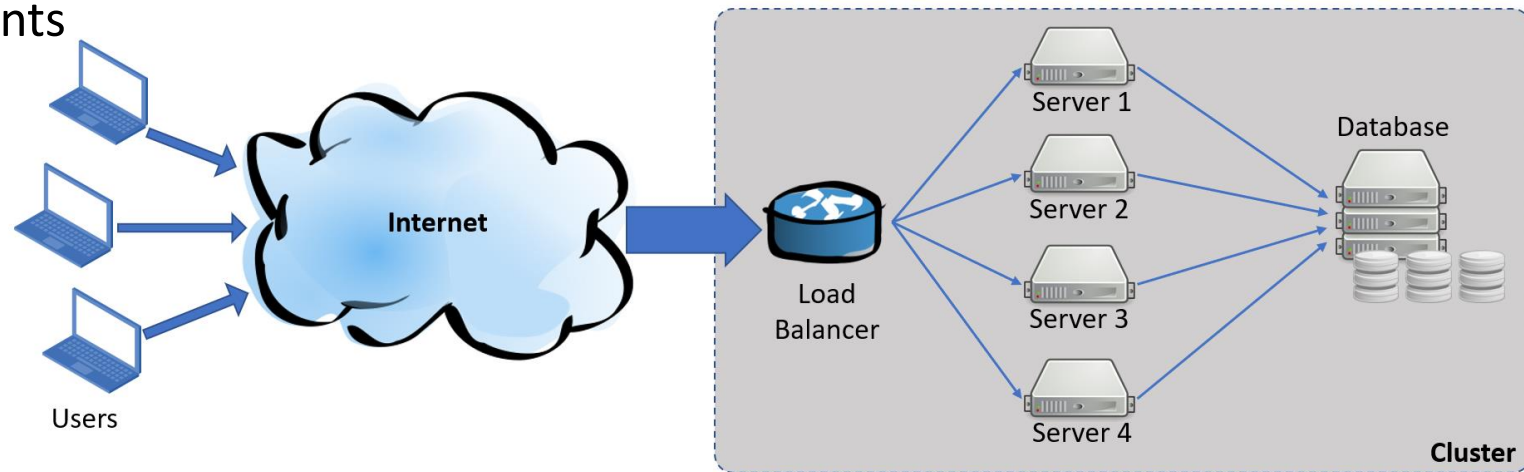
- Usually provided as SaaS
 - Collaborative editing software (Microsoft Word, Google Docs)
 - Cloud storage (Microsoft OneDrive, Dropbox)
 - E-mail (Outlook, Gmail)
- Web or desktop application
- User only cares about using the service
 - Details hidden from the user
 - Management and operation are automated

Examples of automation for end users

- Website creating services
 - Automation of the back-end
 - Web servers
 - Reverse proxies
 - Load balancers
 - Databases
 - Storage
 - Updates to Web components

Examples of automation for end users

- Website creating services
 - Automation of the back-end
 - Web servers
 - Reverse proxies
 - Load balancers
 - Databases
 - Storage
 - Updates to Web components



Examples of automation for end users

- Website creating services
 - Automation of the back-end
 - Web servers
 - Reverse proxies
 - Load balancers
 - Databases
 - Storage
 - Updates to Web components
- E-mail
 - Management of the back-end
 - SMTP, IMAP, and POP servers
 - Blocklists
 - Anti-spam
 - Anti-phishing
 - Data warehousing
 - Web servers for front-end
 - Indexing for search

Automation for enterprises

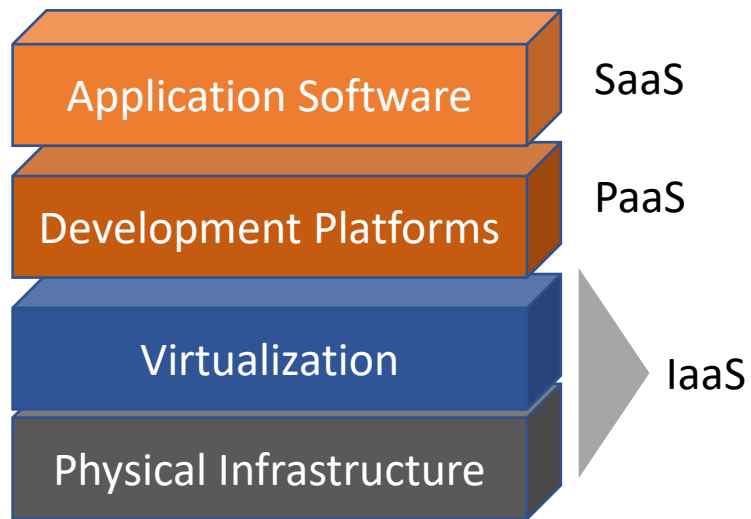
- Enterprises need automation to manage their infrastructure
 - Software deployment and updates
 - Configuring servers and virtual machines
 - Applications on cloud resources
 - Platform as a Service
 - Automation frameworks for other applications
 - Applications on employee devices (e.g., company laptop)

Automation for enterprises

- Enterprises need automation to manage their infrastructure
 - Software deployment and updates
 - Configuring servers and virtual machines
 - Applications on cloud resources
 - Platform as a Service
 - Automation frameworks for other applications
 - Applications on employee devices (e.g., company laptop)
- Frameworks automate computation
 - Hadoop MapReduce, Spark, Kubernetes, and others

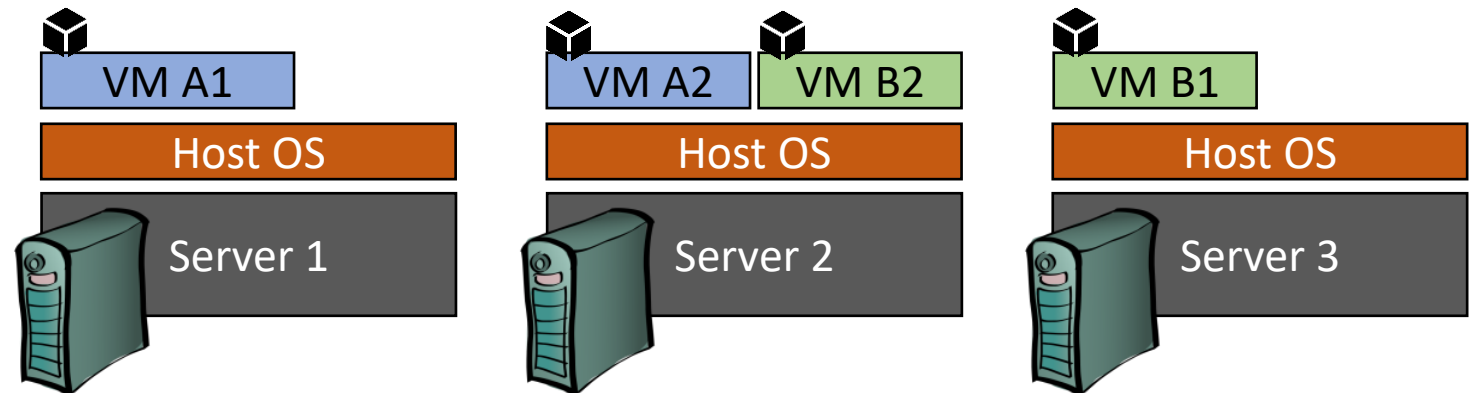
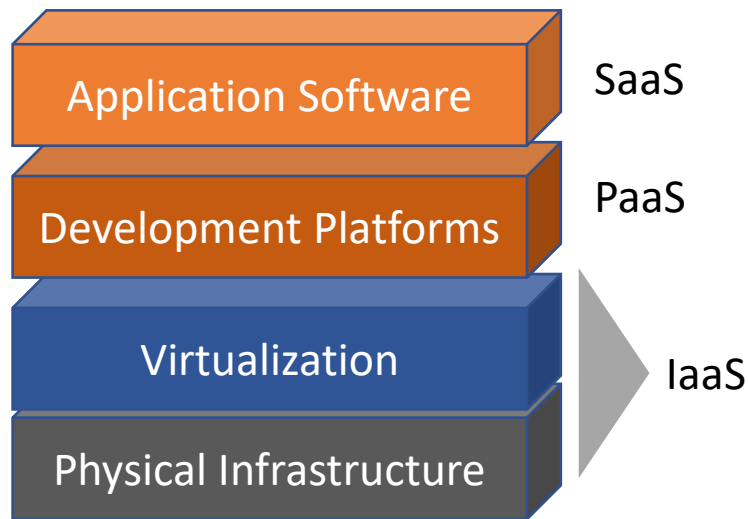
Automation for cloud providers

- Like automation for enterprises, but on a larger scale
- Manage infrastructure
 - Software deployment, configuration, and updates



Automation for cloud providers

- Like automation for enterprises, but on a larger scale
- Manage infrastructure
 - Software deployment, configuration, and updates
- Handling customer demands





CompSci 401: Cloud Computing

Automation in Data Centers

Prof. Ítalo Cunha



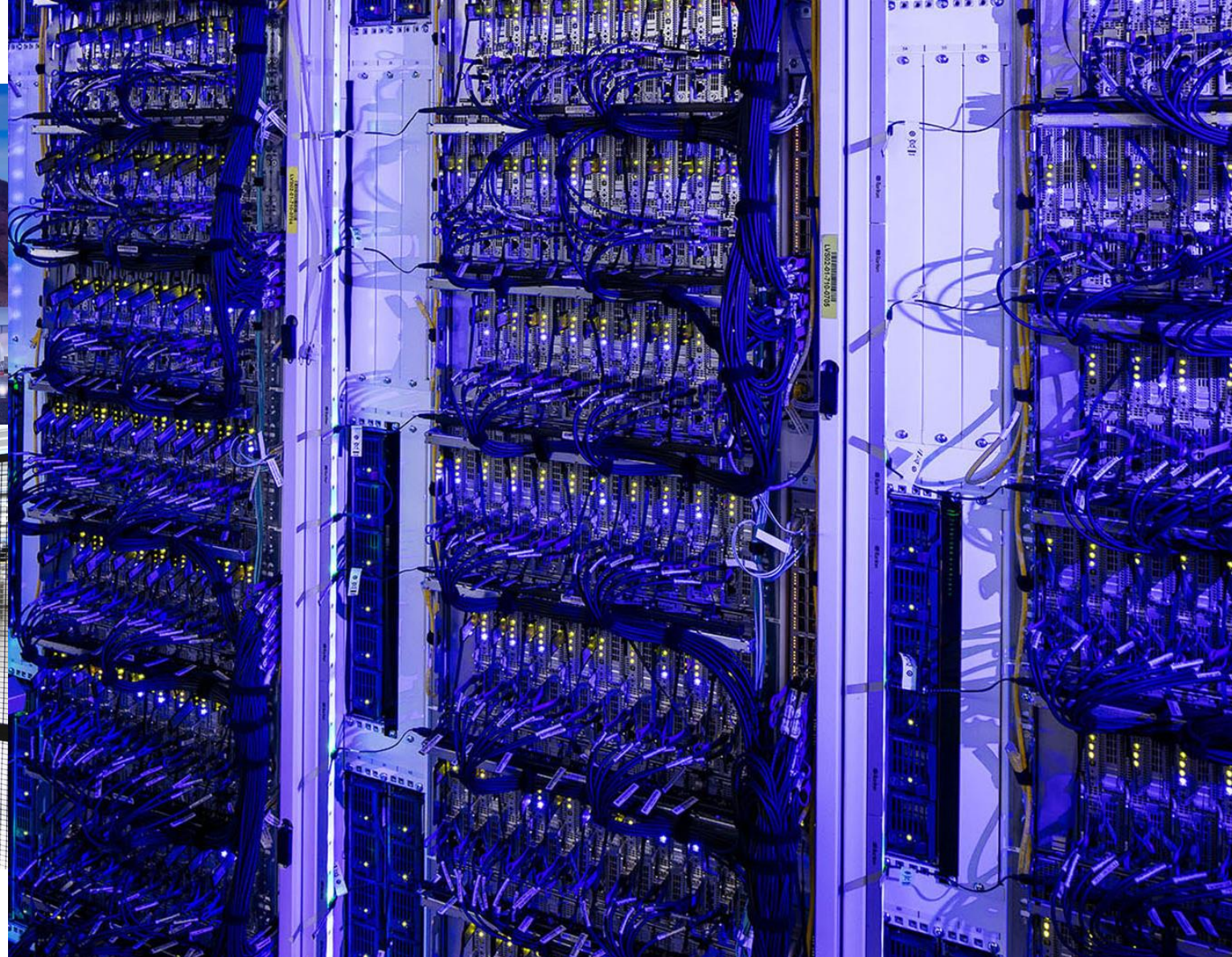
There's a lot on top of the physical infrastructure



There's a lot on top of the physical infrastructure



There's a lot on top of the physical infrastructure





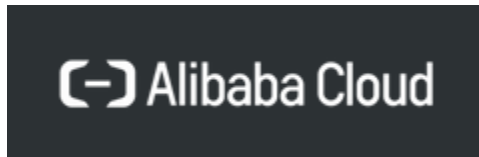
Servers, switches, routers

- Operating system/hypervisor
 - Configuration
 - User credentials
 - Remote services
 - Applications
 - Management
 - Monitoring
- Network
 - DNS resolver
 - IP addresses
 - Routes



Need for automation in data centers

- Extreme scale
 - Thousands of network devices
 - Hundreds of thousands of servers
 - Thousands of customers, some of them large enterprises
 - Millions of services
- Diverse services
 - Large cloud providers have several offerings



Elastic Computing

Networking & CDN

Database

Storage

Security

Enterprise Applications & Cloud
Communication

Analytics

Artificial Intelligence

Media Services

Hybrid Cloud

Container & Middleware

Developer Services

Internet of Things

Alibaba Cloud Academy

Management

Compute

Storage

Operations

Networking

Big Data

Databases

Serverless

Security

CI/CD

Application Integration

Artificial Intelligence

Tools

Other Google products

Support

Recently visited

Favorites

All services

- Analytics
- Application Integration
- AR & VR
- AWS Cost Management
- Blockchain
- Business Applications
- Compute
- Containers
- Customer Enablement
- Database
- Developer Tools
- End User Computing
- Front-end Web & Mobile
- Game Development
- Internet of Things
- Machine Learning
- Management & Governance
- Media Services
- Migration & Transfer
- Networking & Content Delivery
- Quantum Technologies
- Robotics
- Satellite
- Security, Identity, & Compliance
- Storage

Need for automation

- Extreme scale
 - Thousands of network devices
 - Hundreds of thousands of servers
 - Thousands of customers
 - Millions of services
- Diverse services
 - Large cloud providers have several offerings

FEATURED PRODUCTS

Droplets

Scalable virtual machines

Kubernetes

Managed Kubernetes clusters

App Platform

Get apps to market faster

Databases

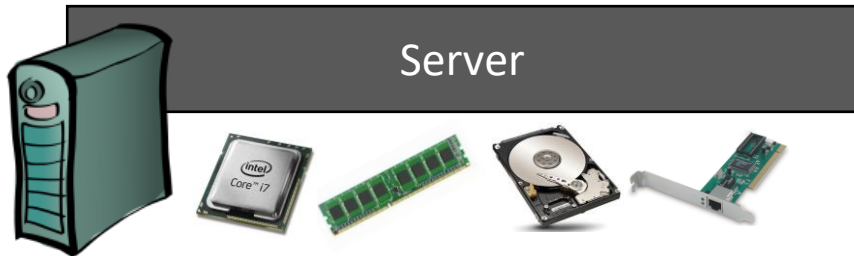
Worry-free setup & maintenance

Spaces

Simple object storage

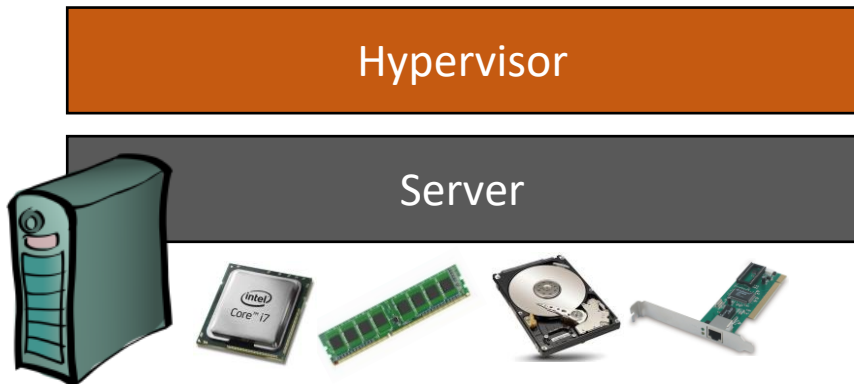
Example: VM deployment

- Choose a server where to run the VM



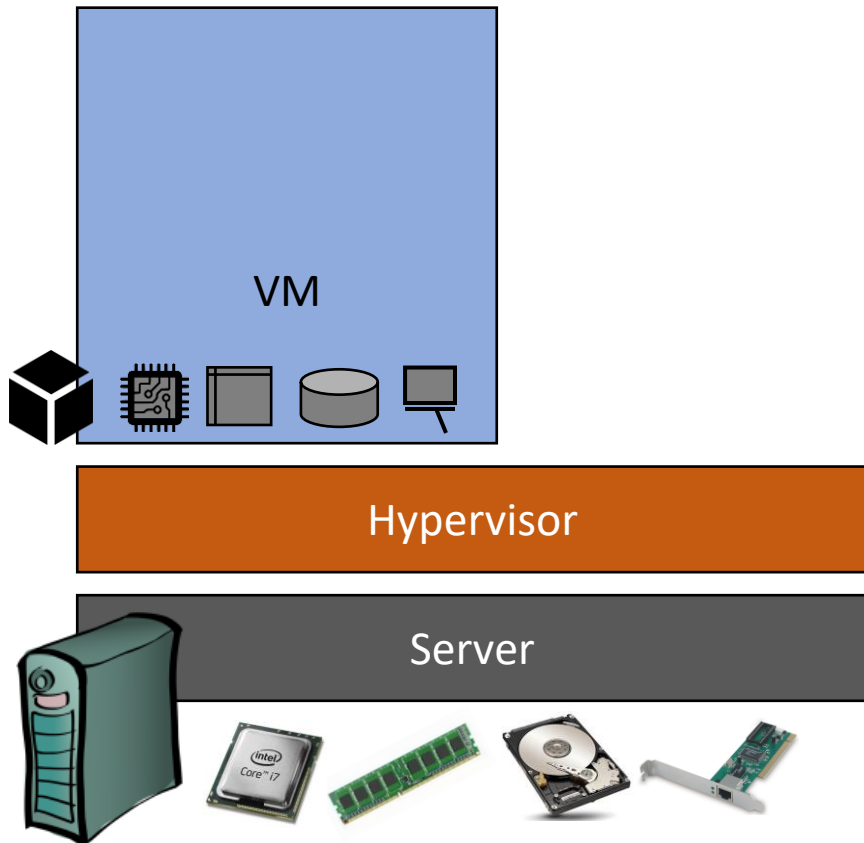
Example: VM deployment

- Choose a server where to run the VM
- Assume that the hypervisor is configured 😊



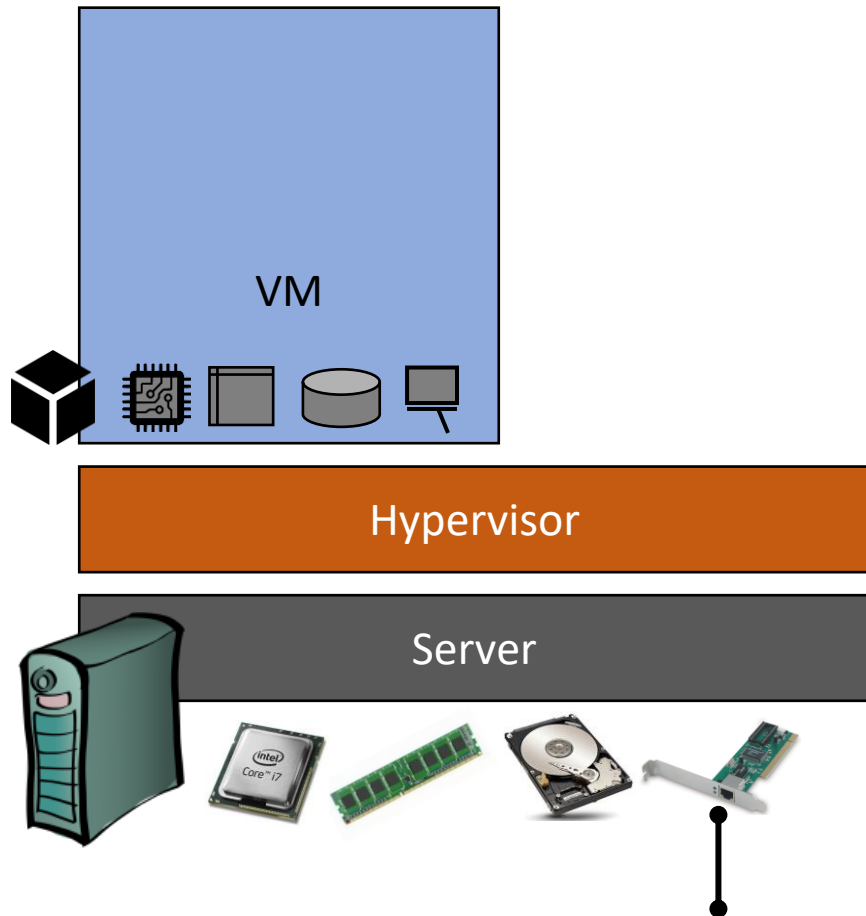
Example: VM deployment

- Choose a server where to run the VM
- Assume that the hypervisor is configured 😊
- Configure the hypervisor to run the VM



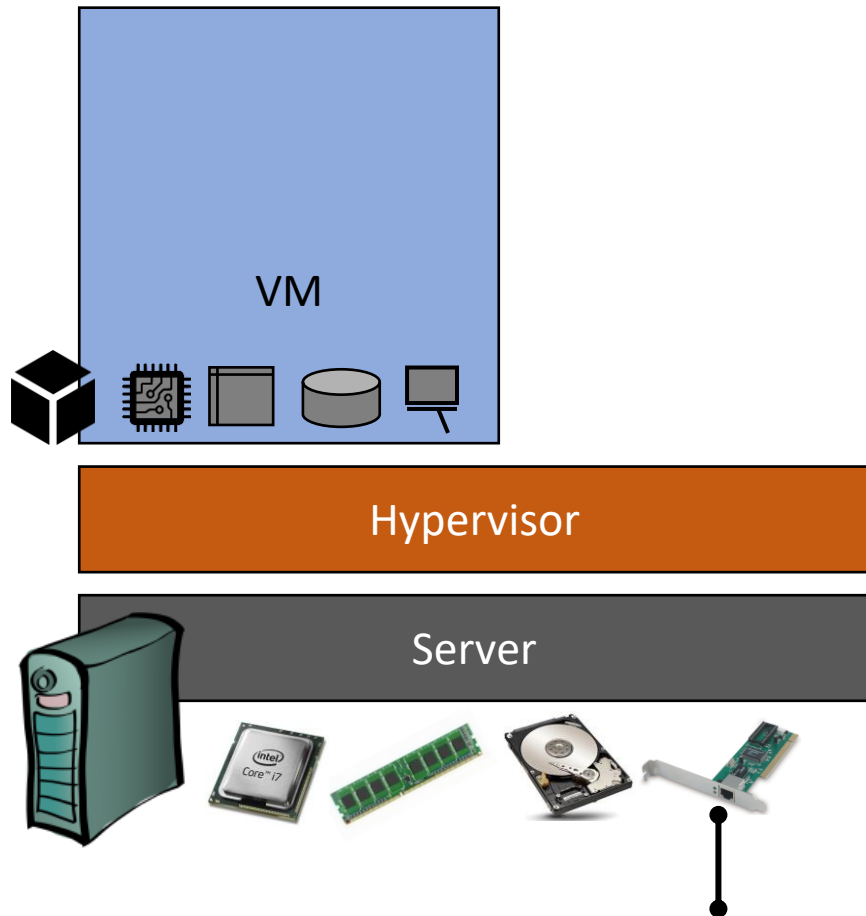
Example: VM deployment

- Choose a server where to run the VM
- Assume that the hypervisor is configured 😊
- Configure the hypervisor to run the VM
- Configure the network to forward packets
 - Possibly involving a tenant's virtual network



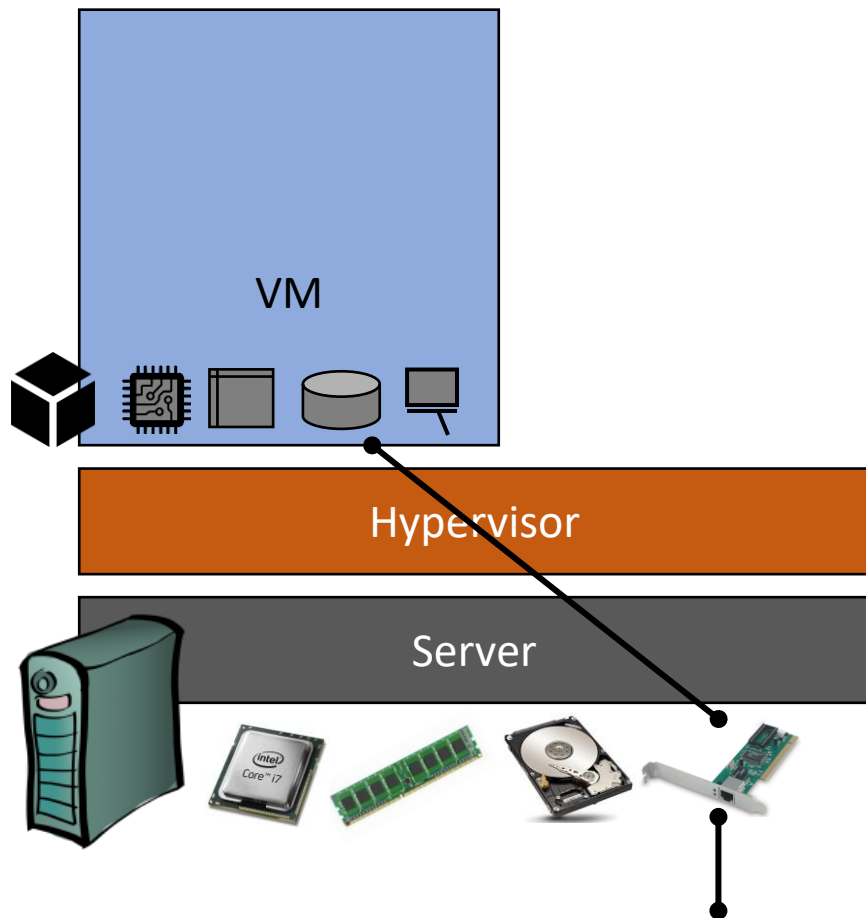
Example: VM deployment

- Choose a server where to run the VM
- Assume that the hypervisor is configured 😊
- Configure the hypervisor to run the VM
- Configure the network to forward packets
 - Possibly involving a tenant's virtual network
- Choose remote disk server where to store the VM disk image

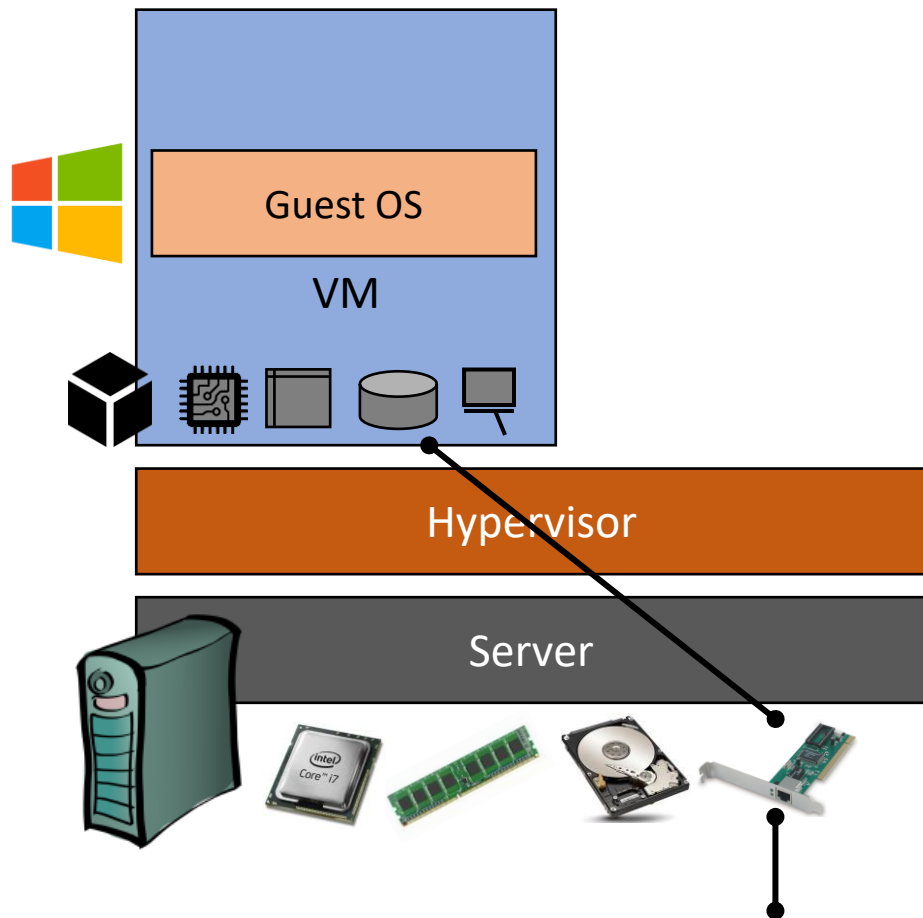


Example: VM deployment

- Choose a server where to run the VM
- Assume that the hypervisor is configured 😊
- Configure the hypervisor to run the VM
- Configure the network to forward packets
 - Possibly involving a tenant's virtual network
- Choose remote disk server where to store the VM disk image
- Configure virtual disk to forward disk accesses

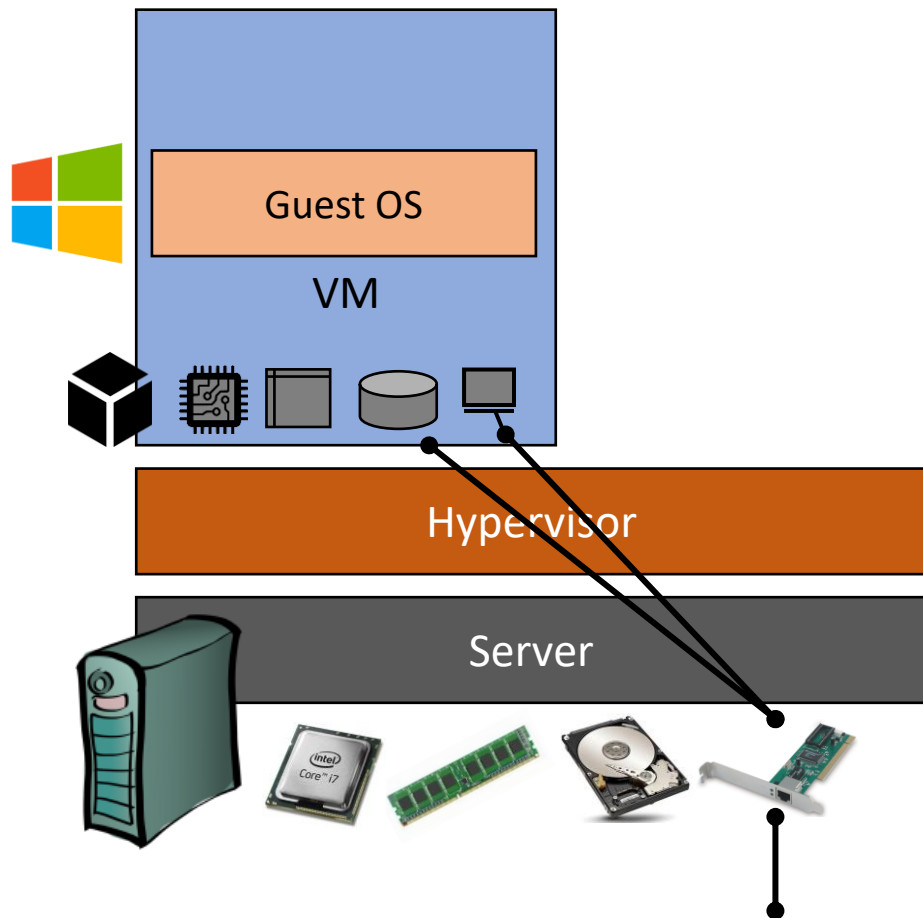


Example: VM deployment



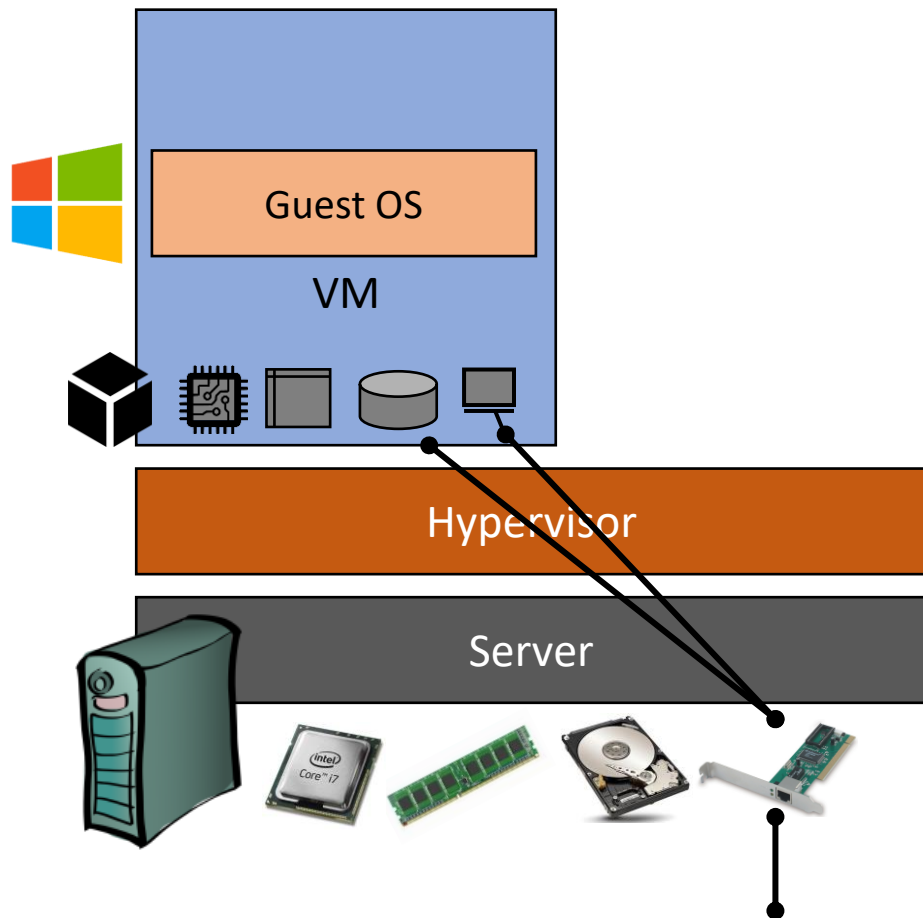
- Choose a server where to run the VM
- Assume that the hypervisor is configured 😊
- Configure the hypervisor to run the VM
- Configure the network to forward packets
 - Possibly involving a tenant's virtual network
- Choose remote disk server where to store the VM disk image
- Configure virtual disk to forward disk accesses
- Create disk image with guest OS

Example: VM deployment



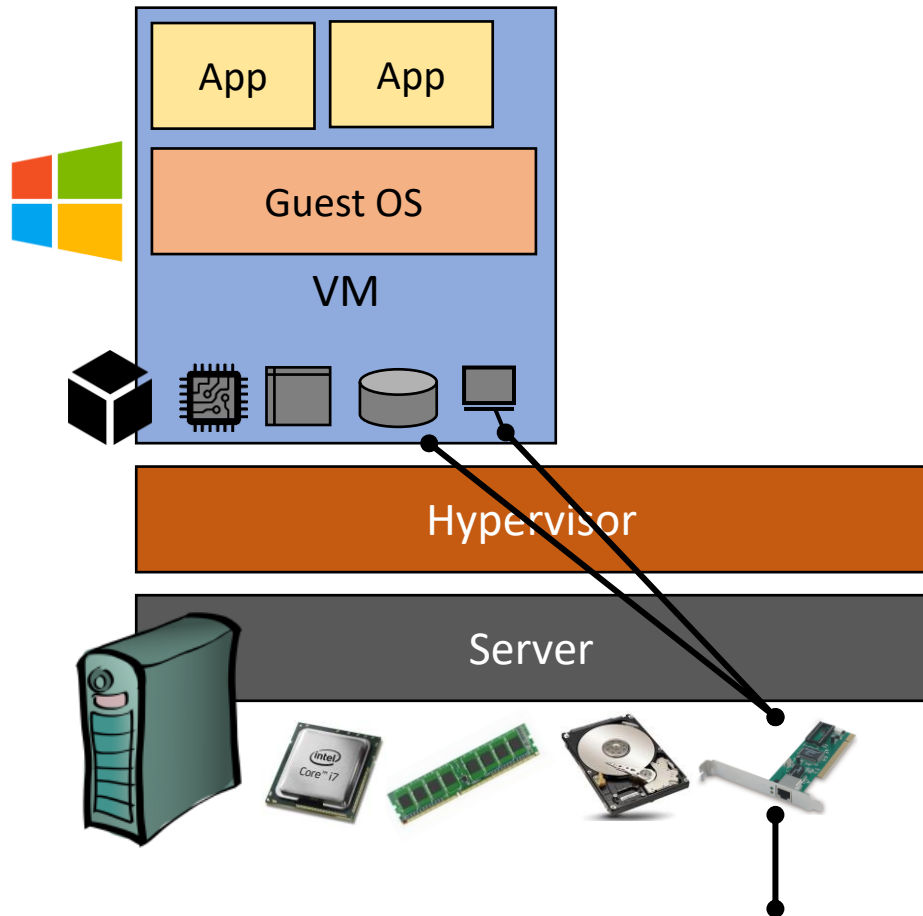
- Choose a server where to run the VM
- Assume that the hypervisor is configured 😊
- Configure the hypervisor to run the VM
- Configure the network to forward packets
 - Possibly involving a tenant's virtual network
- Choose remote disk server where to store the VM disk image
- Configure virtual disk to forward disk accesses
- Create disk image with guest OS
- Configure IP address on the VM so it's accessible

Example: VM deployment



- Choose a server where to run the VM
- Assume that the hypervisor is configured 😊
- Configure the hypervisor to run the VM
- Configure the network to forward packets
 - Possibly involving a tenant's virtual network
- Choose remote disk server where to store the VM disk image
- Configure virtual disk to forward disk accesses
- Create disk image with guest OS
- Configure IP address on the VM so it's accessible
- Create credentials on the VM for customer access

Example: VM deployment



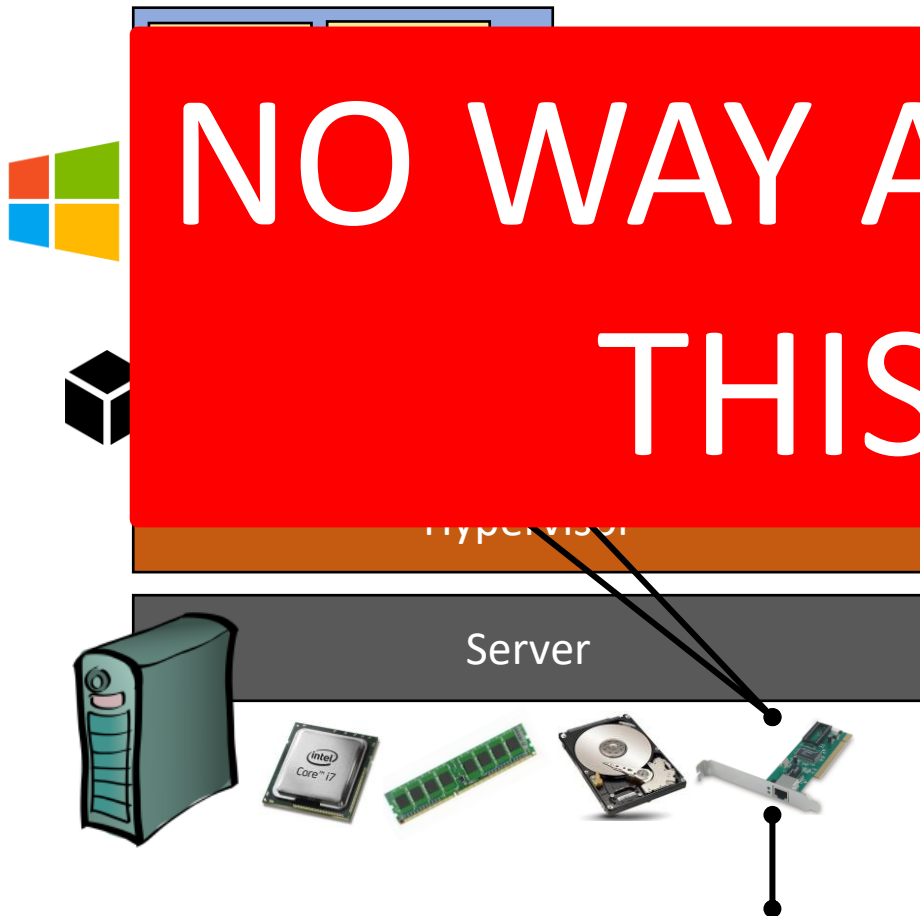
- Choose a server where to run the VM
- Assume that the hypervisor is configured 😊
- Configure the hypervisor to run the VM
- Configure the network to forward packets
 - Possibly involving a tenant's virtual network
- Choose remote disk server where to store the VM disk image
- Configure virtual disk to forward disk accesses
- Create disk image with guest OS
- Configure IP address on the VM so it's accessible
- Create credentials on the VM for customer access
- Possibly install and configure applications requested by the user

Example: VM deployment

- Choose a server where to run the VM
- Assume that the hypervisor is configured 😊

**NO WAY ANYONE IS DOING
THIS MANUALLY**

- Configure IP address on the VM so it's accessible
- Create credentials on the VM for customer access
- Possibly install and configure applications requested by the user



Need for automation in data centers

- Extreme scale
 - Thousands of network devices
 - Hundreds of thousands of servers
 - Thousands of customers
 - Millions of services
- Diverse services
 - Large cloud providers have several offerings
- Constant change
 - Customers, technologies, practices, and requirements change over time
 - Customers can request services at any time, and expect immediate provisioning



CompSci 401: Cloud Computing

What Can Be Automated?

Prof. Ítalo Cunha



What can be automated?

- Most tasks related to software and data can be automated
- Run as much as possible in software
 - Virtualization
 - Software-defined networks
- “Lights-out” datacenter: remote automated management

Management of virtual resources

- Creation, deletion, modification, and reconfiguration of
 - Virtual machines
 - Virtual disks
 - Virtual networks
 - Containers
 - Applications
- Autoscaling

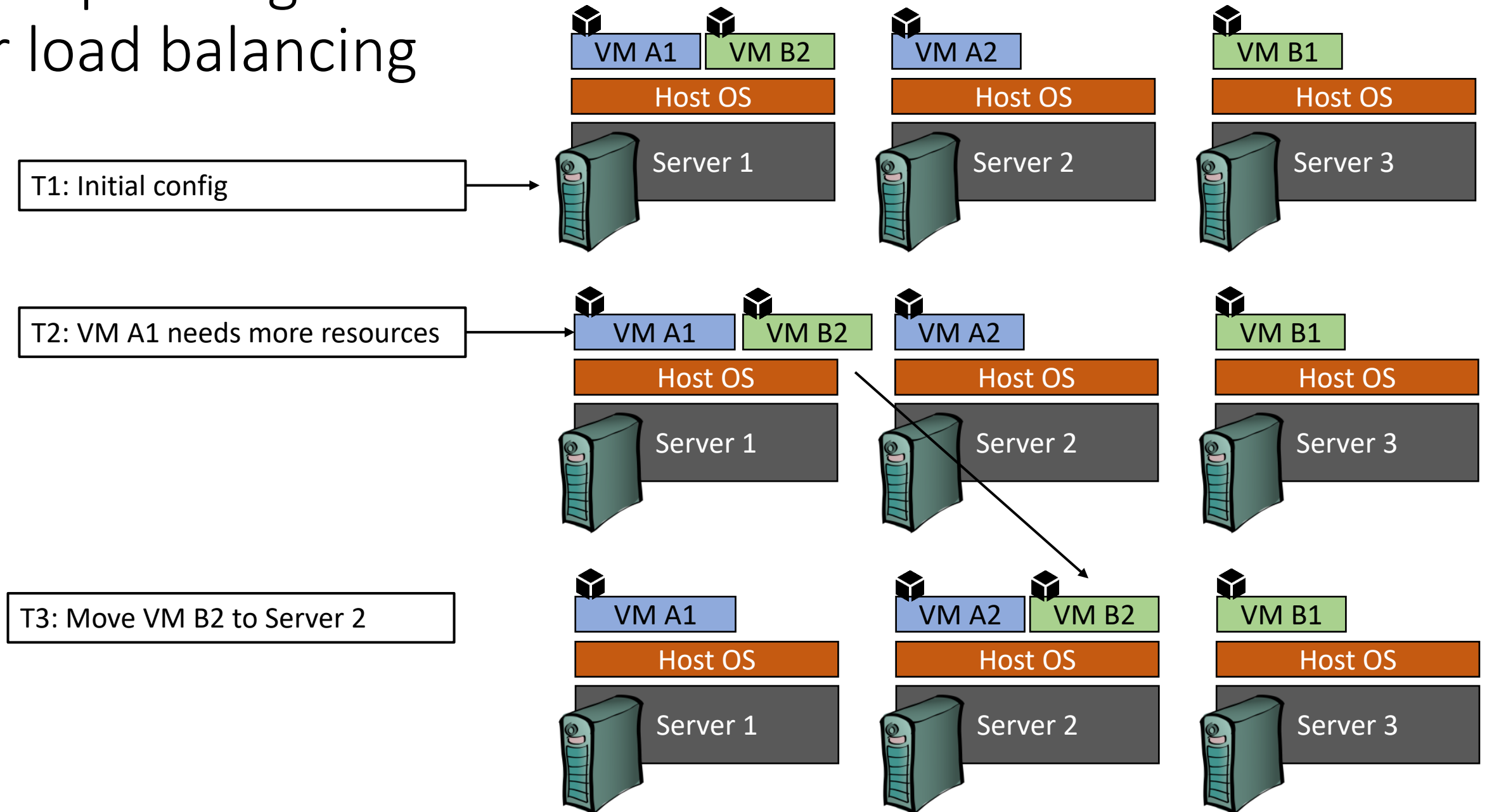
Workload monitoring and accounting

- Cloud providers measure *a lot*
- Billing customers
- Storing and indexing monitoring information useful for troubleshooting
 - Server and network failures, hotspots, security issues
- Identify trends and projecting demand
 - What resources are in high demand
 - When to expand capacity

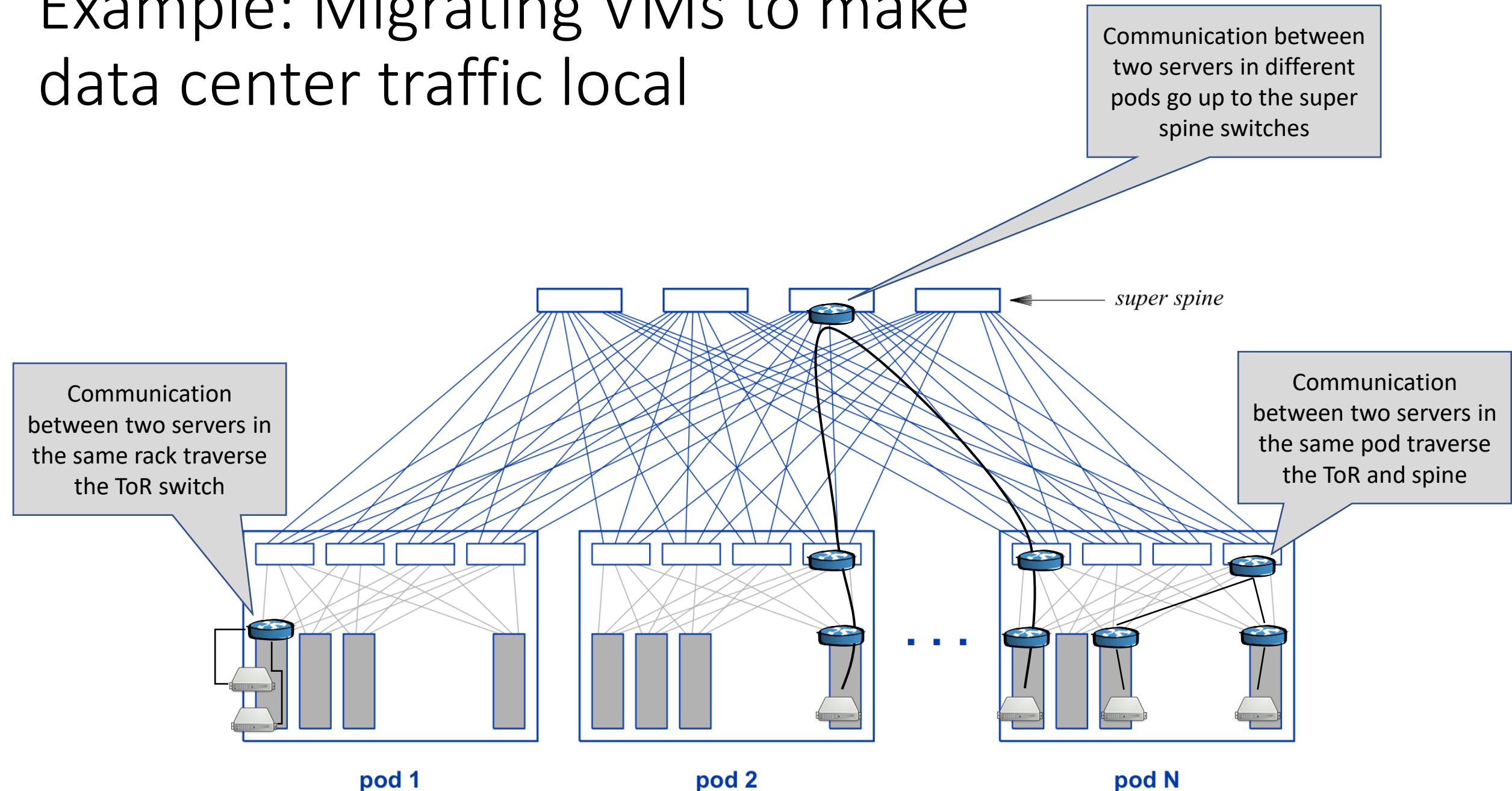
Optimization

- Reconfiguration and placement of virtual resources
 - Load balancing across servers
 - Improve performance
 - Moving virtual resources from underutilized physical resources
 - Save power and reduce wear
- Network traffic localization
 - Reduce link utilization and end-to-end latency
 - Server-to-server and network storage

Example: Migration for load balancing



Example: Migrating VMs to make data center traffic local



Safety and recovery

- Scheduled or periodic backup of customer data
- Data replication for performance and resilience
- Fast rerouting around failed network devices
- Failure detection from monitoring data
- Anomaly detection from monitoring data
 - Partially malfunctioning devices
 - Buggy applications
 - Misbehaving VMs
 - Security breaches and attacks

Software updates

- Keeping applications and the operating system up-to-date
 - Latest features
 - Better performance
 - Security issues
- Upgrading software installed by customers
- Providing tooling for continuous deployment and integration

Security policy enforcement

- Applying security patches to fix vulnerabilities
- Configuration of firewalls and access control lists (ACLs)
- Manage secrets, certificates, and encryption keys
- Configuring virtual private networks (VPNs) and encryption



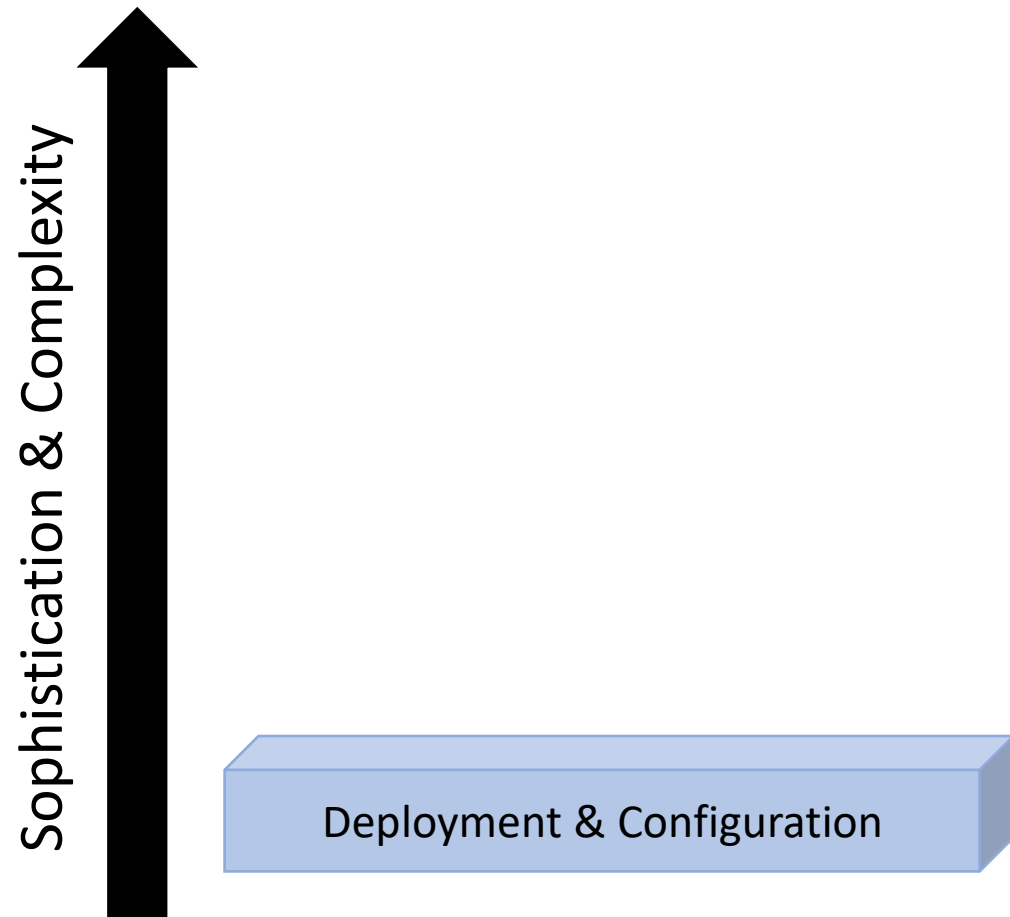
CompSci 401: Cloud Computing

Levels of Automation

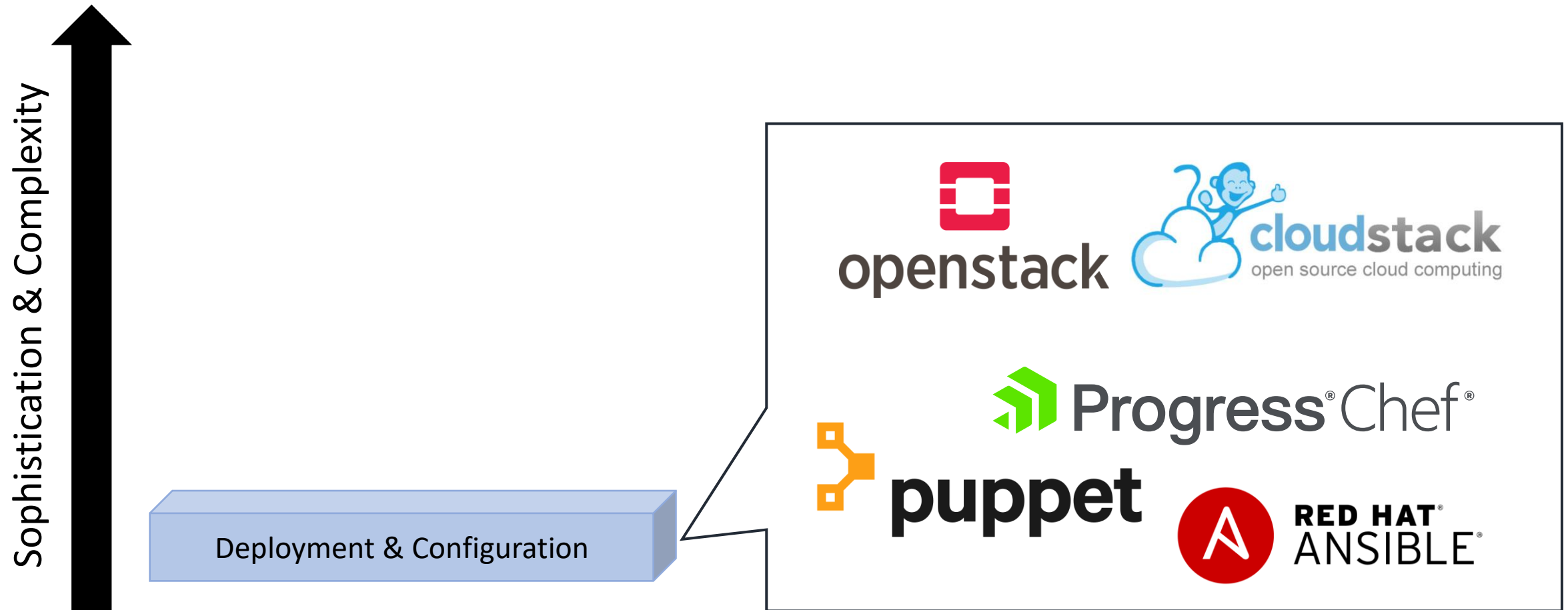
Prof. Ítalo Cunha



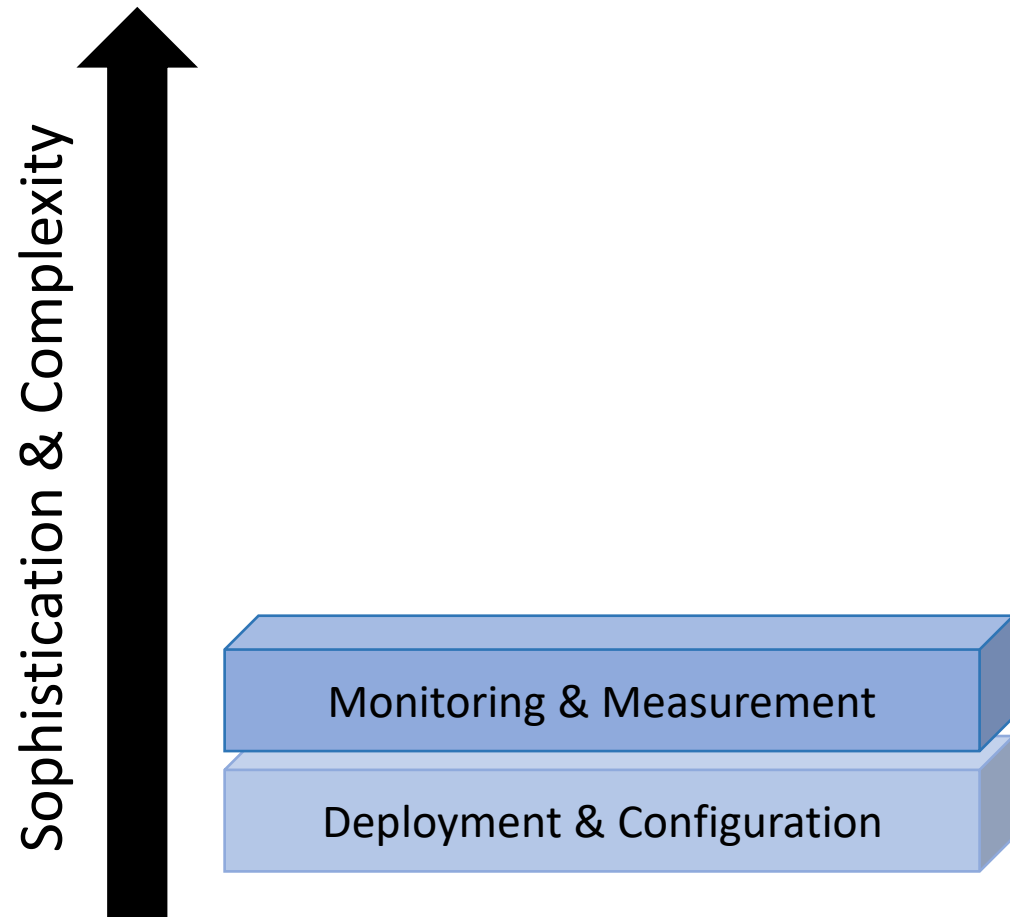
Levels of Automation



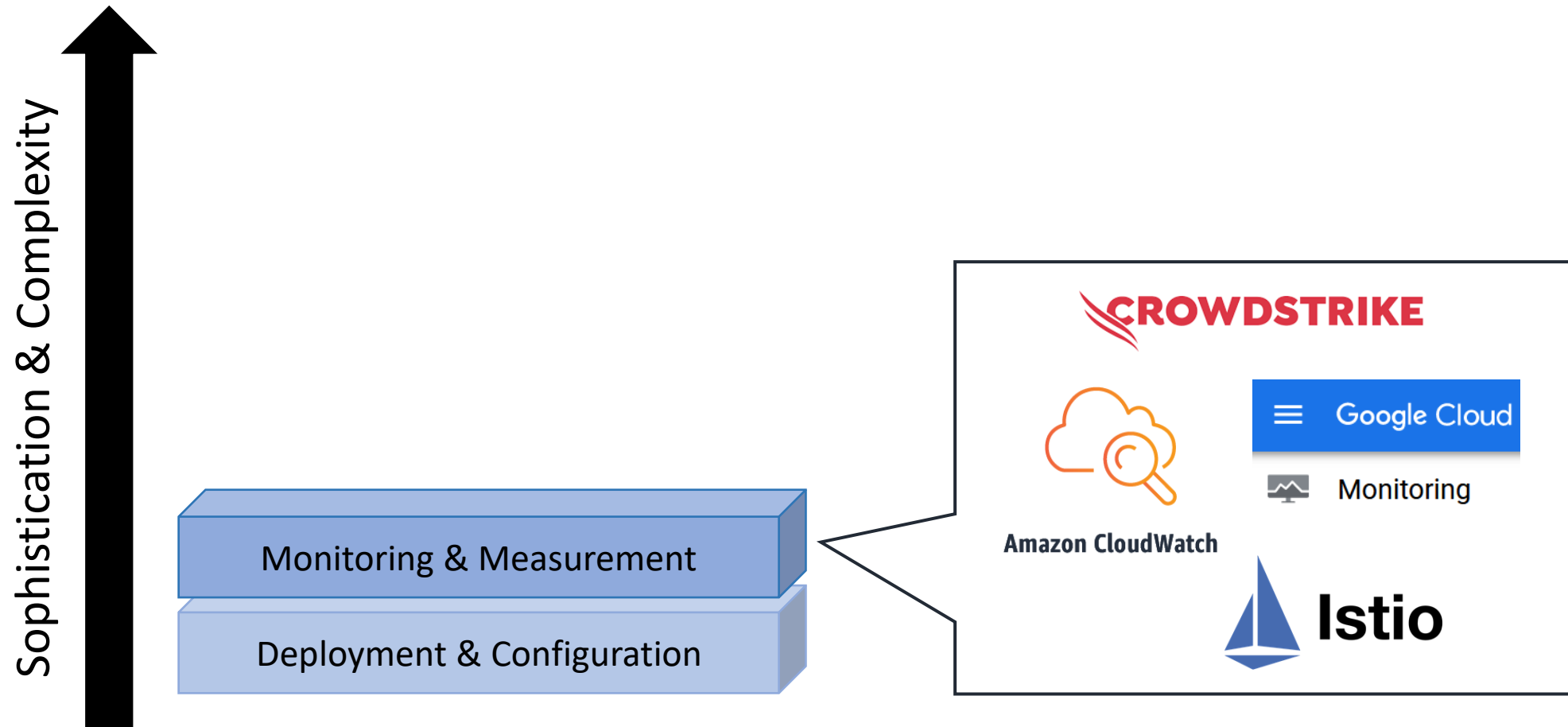
Levels of Automation



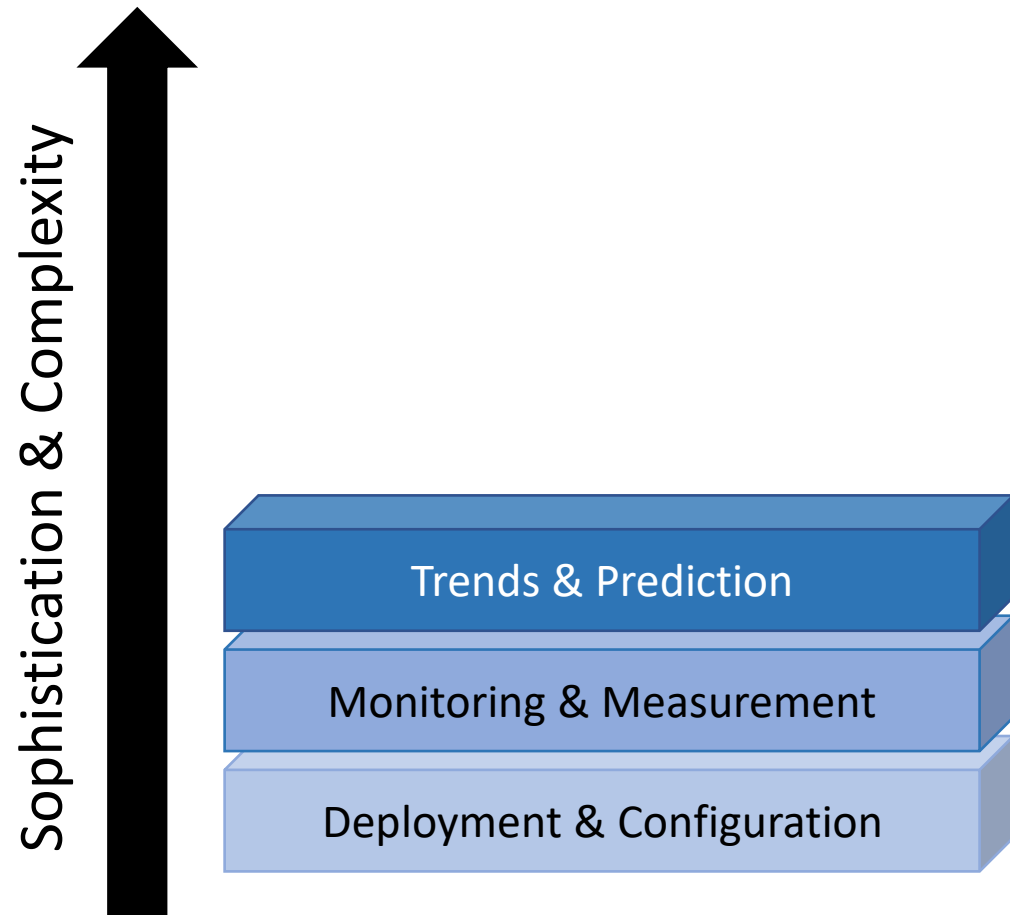
Levels of Automation



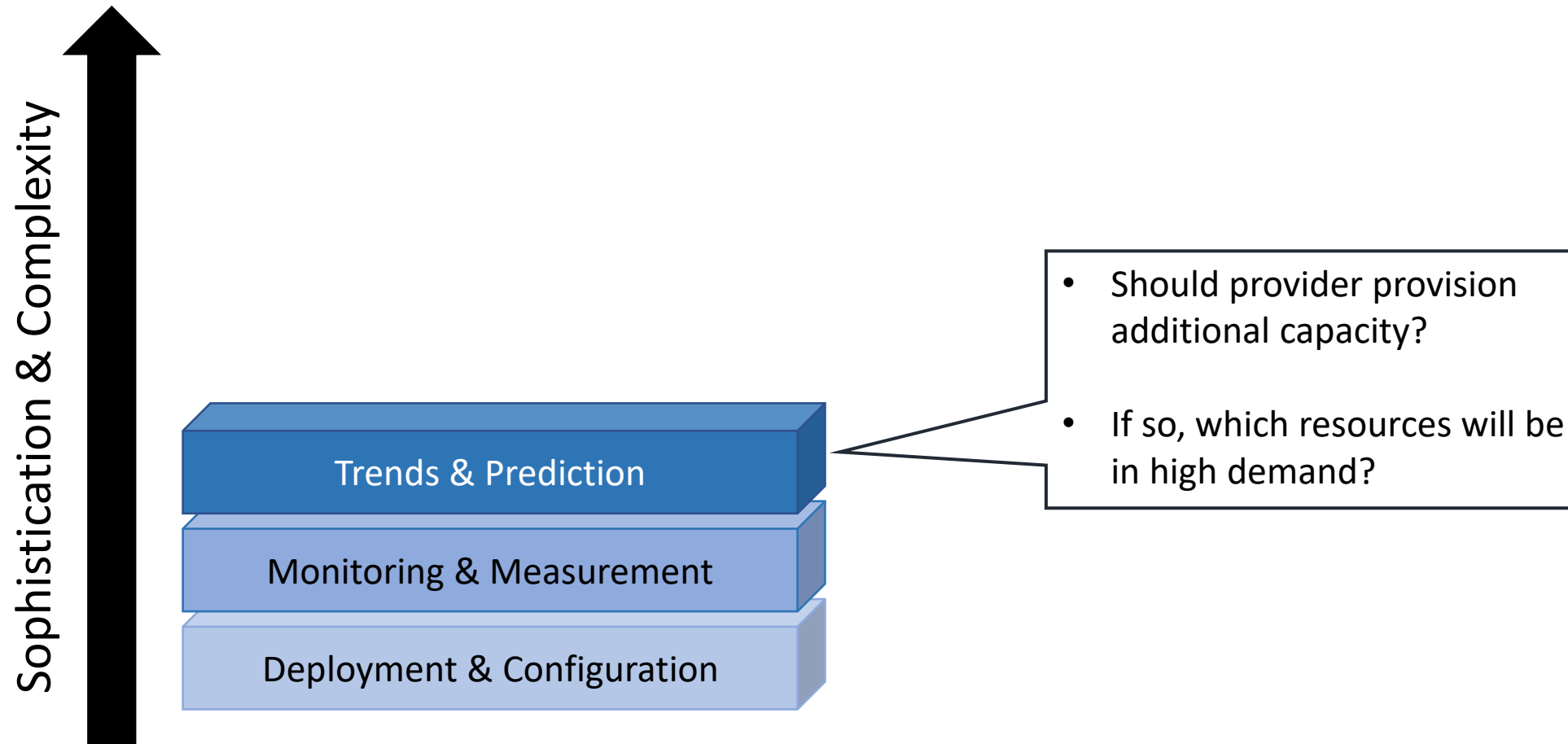
Levels of Automation



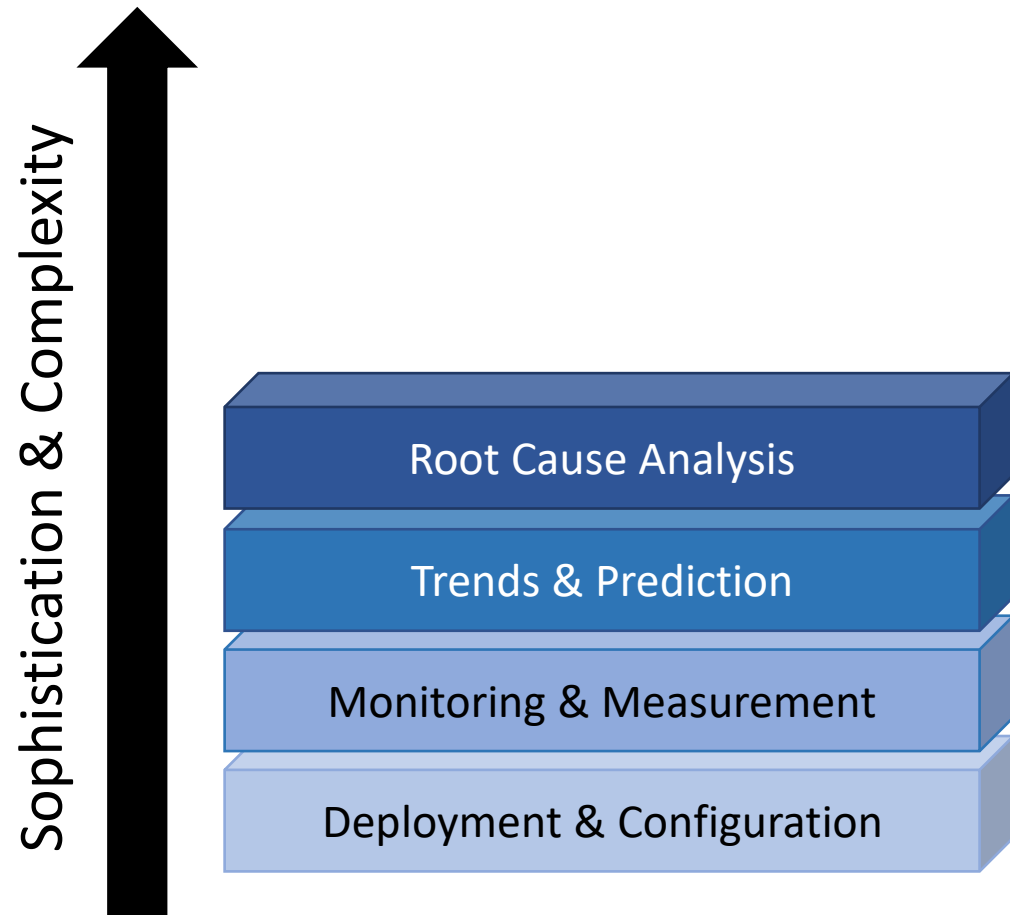
Levels of Automation



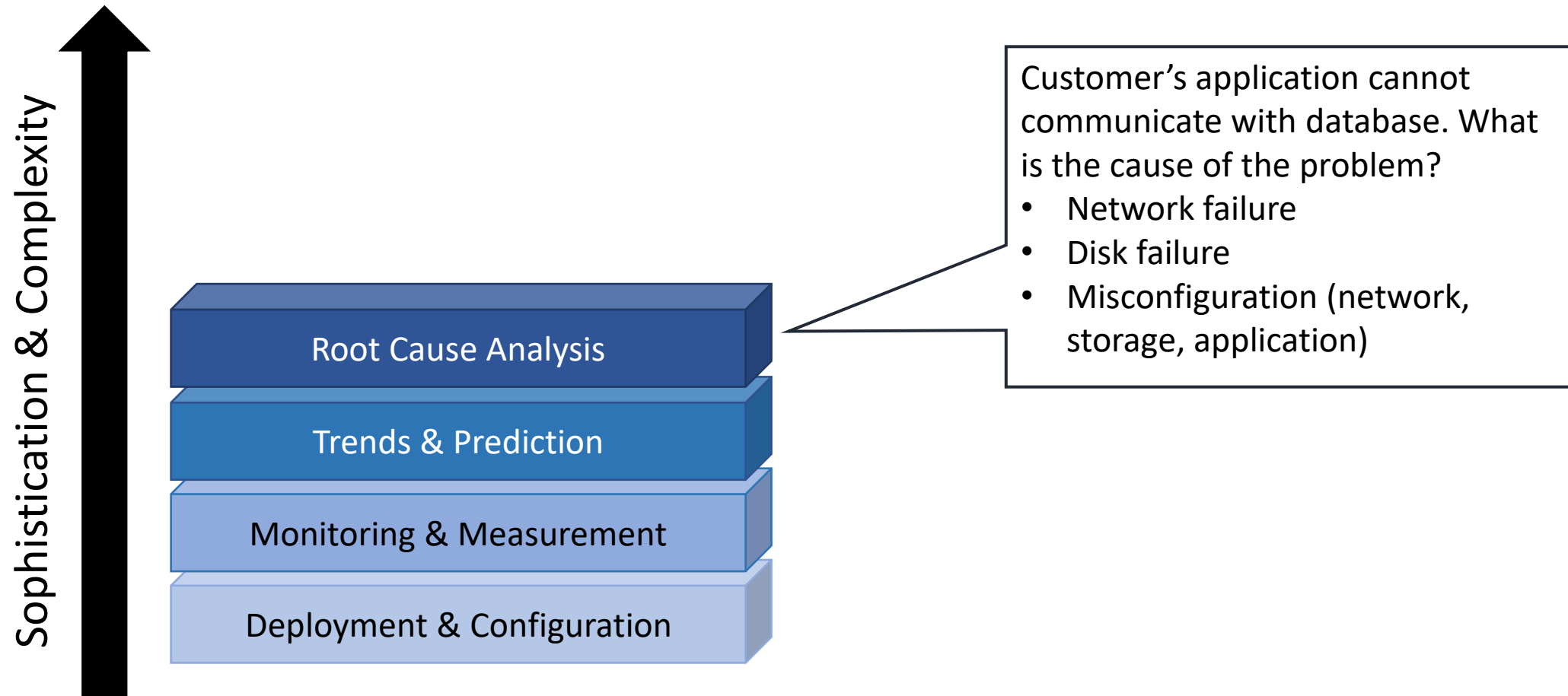
Levels of Automation



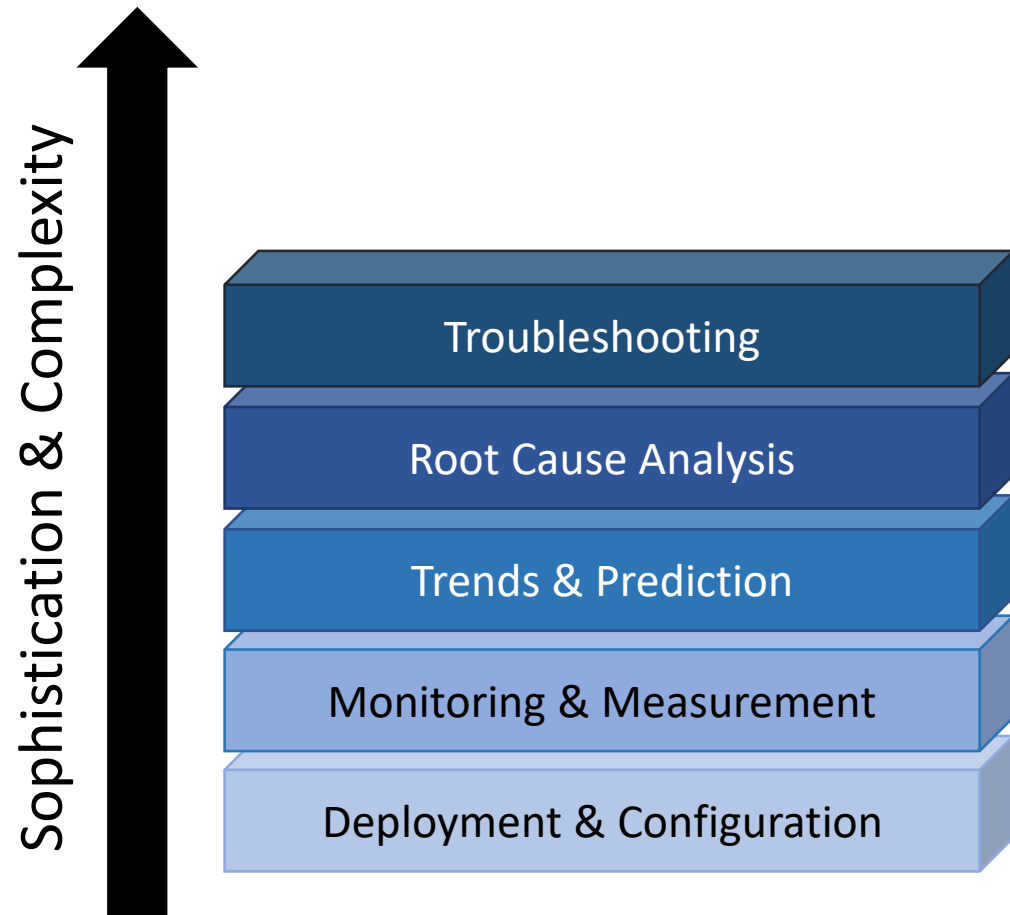
Levels of Automation



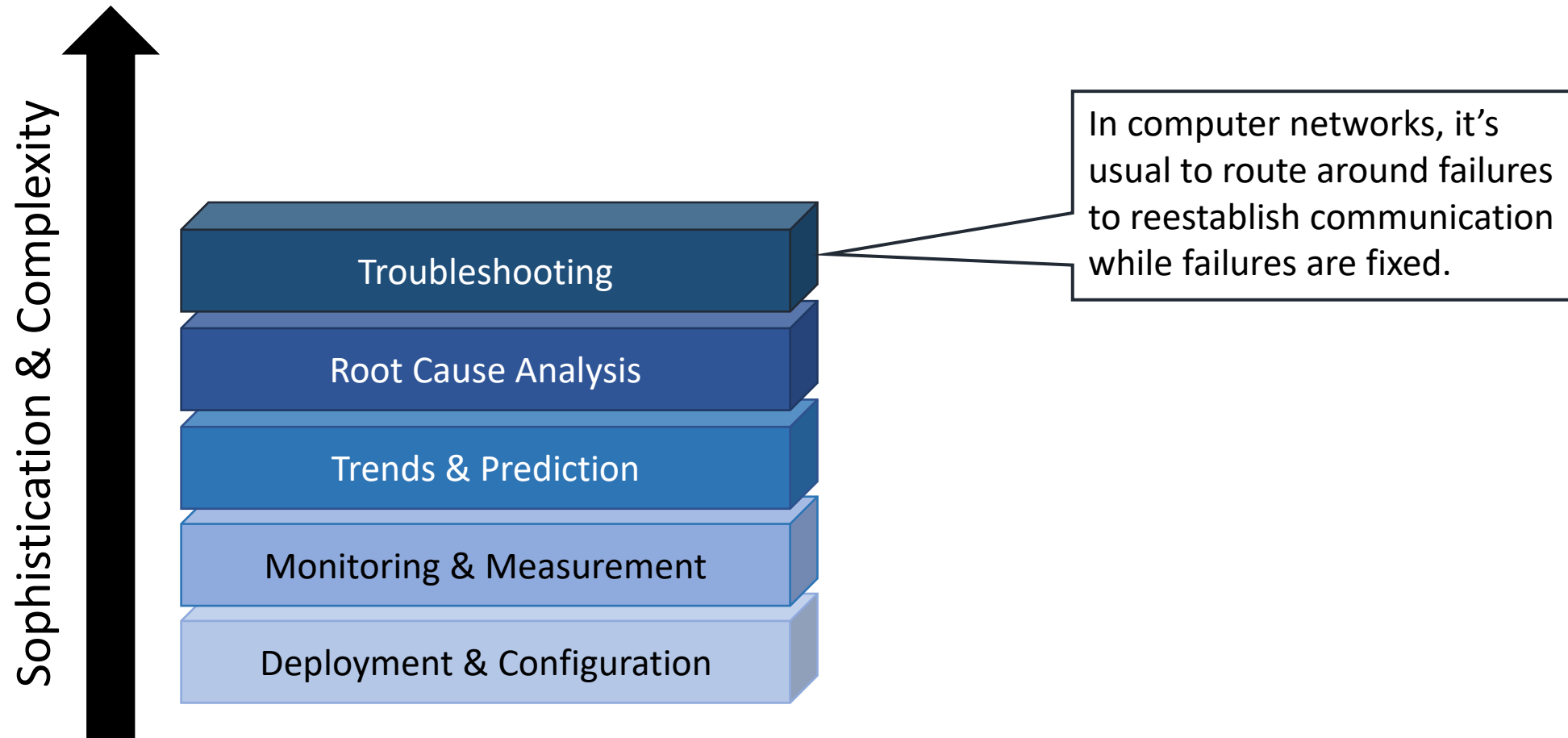
Levels of Automation



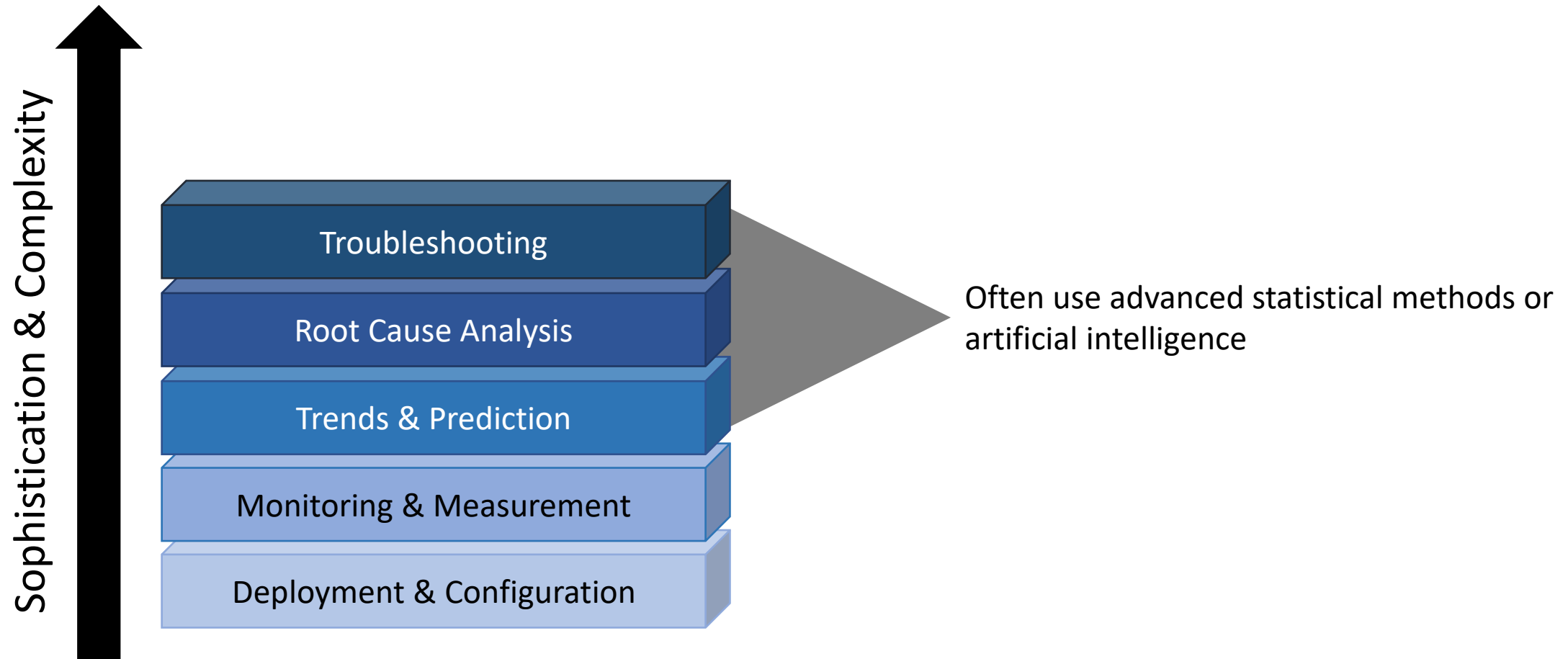
Levels of Automation



Levels of Automation



Levels of Automation





CompSci 401: Cloud Computing

Automation Tools

Prof. Ítalo Cunha



Lots of automation tools

- Automation is great, so let's have more of it
- Computing infrastructures are complex
 - Many, possibly conflicting operational goals
 - Keep VMs of a single tenant close together
 - Keep VMs on lightly-loaded servers
 - Tools reflect these different goals

Lots of automation tools

- Automation is great, so let's have more of it
- Computing infrastructures are complex
 - Many, possibly conflicting operational goals
 - Keep VMs of a single tenant close together
 - Keep VMs on lightly-loaded servers
 - Tools reflect these different goals
- Tools to automate one or a subset of tasks
 - Operators often specialize in specific tasks
 - Tools can capture this knowledge

Automating configuration across vendors

- Configuration is key, and a major target for automation
- But different vendors have different configuration mechanisms
 - Similar functionalities, but different configuration

Networking



Linux Distros



Web Servers



Automating configuration across vendors

- Configuration is key, and a major target for automation
- But different vendors have different configuration mechanisms
 - Similar functionalities, but different configuration
- Vendor-independent configuration language
 - Tool translates configuration into each vendor's specific format
 - Operators does not need to interact with the system being configured

Automating configuration across vendors

- Configuration is key, and a major target for automation
- But different vendors have different configuration mechanisms
 - Similar functionalities, but different configuration
- Vendor-independent configuration language
 - Tool translates configuration into each vendor's specific format
 - Operators does not need to interact with the system being configured

Example:



```
tasks:
- name: Create user
  become: yes
  user:
    name: "{{ item }}"
    groups: hadoop,docker
    state: present
    system: no
    createhome: yes
  with_items: "{{ users }}"
```



Declarative vs Imperative Specification Languages

- *Imperative*: State action to be performed
 - Create user cunha on all virtual machines
 - Assign IP address 192.168.1.17 to the VM's Ethernet interface
- *Declarative*: State the desired outcome
 - User cunha should exist on all virtual machines
 - The VM's main IP address is 192.168.1.17

Declarative vs Imperative Specification Languages

- *Imperative*: State action to be performed
 - Create user cunha on all virtual machines
 - Assign IP address 192.168.1.17 to the VM's Ethernet interface
- *Declarative*: State the desired outcome
 - User cunha should exist on all virtual machines
 - The VM's main IP address is 192.168.1.17
- Complications arise when specification does not cover corner cases
 - What if user cunha already exists?
 - What if the VM has no Ethernet interface or multiple interfaces?

Example:



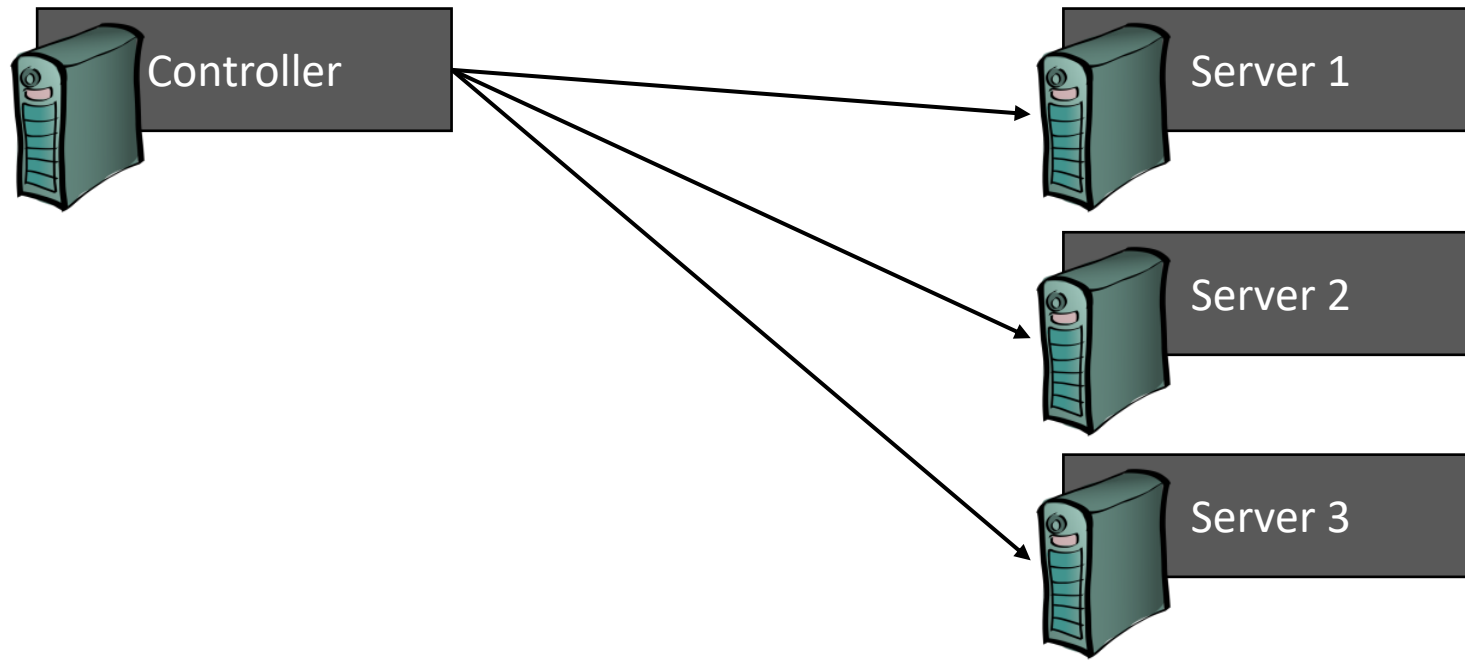
```
tasks:
- name: Create user
  become: yes
  user:
    name: "{{ item }}"
    groups: hadoop,docker
    state: present
    system: no
    createhome: yes
  with_items: "{{ users }}"
```

Declarative
specification



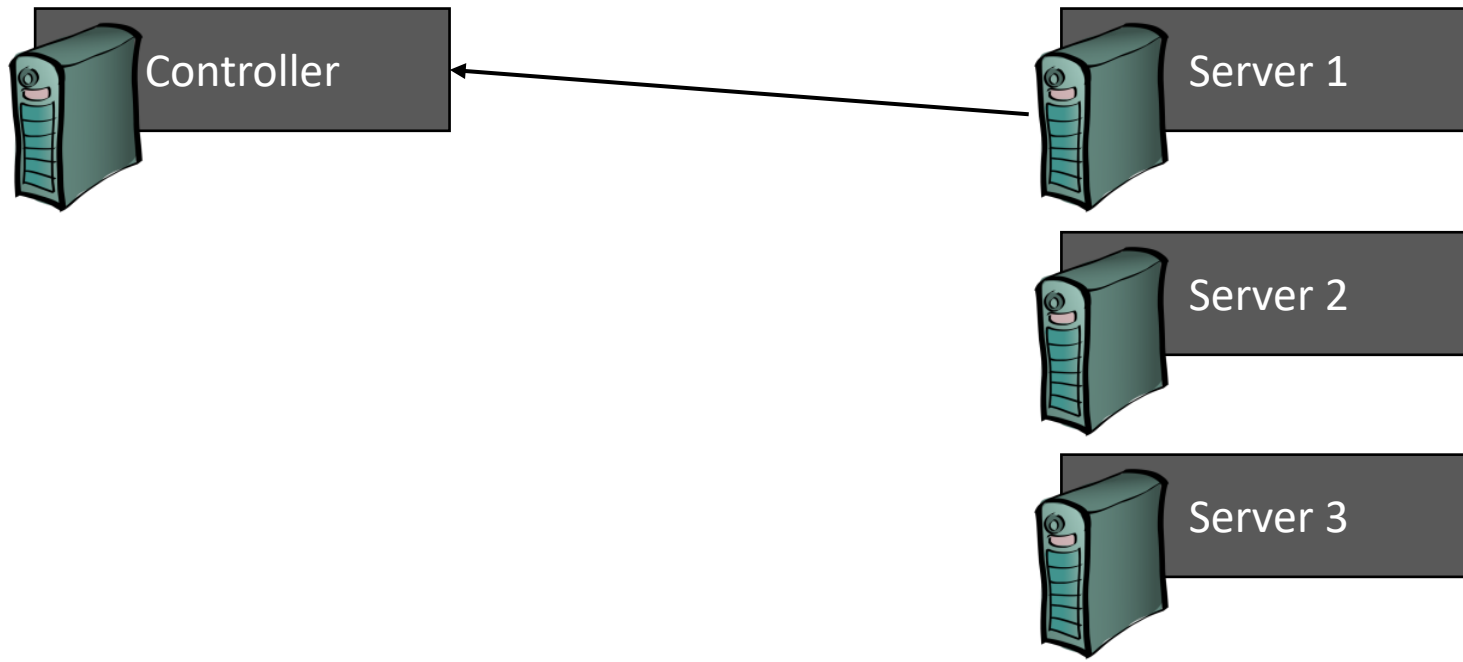
Push vs Pull

- Push → Central controller configures all devices



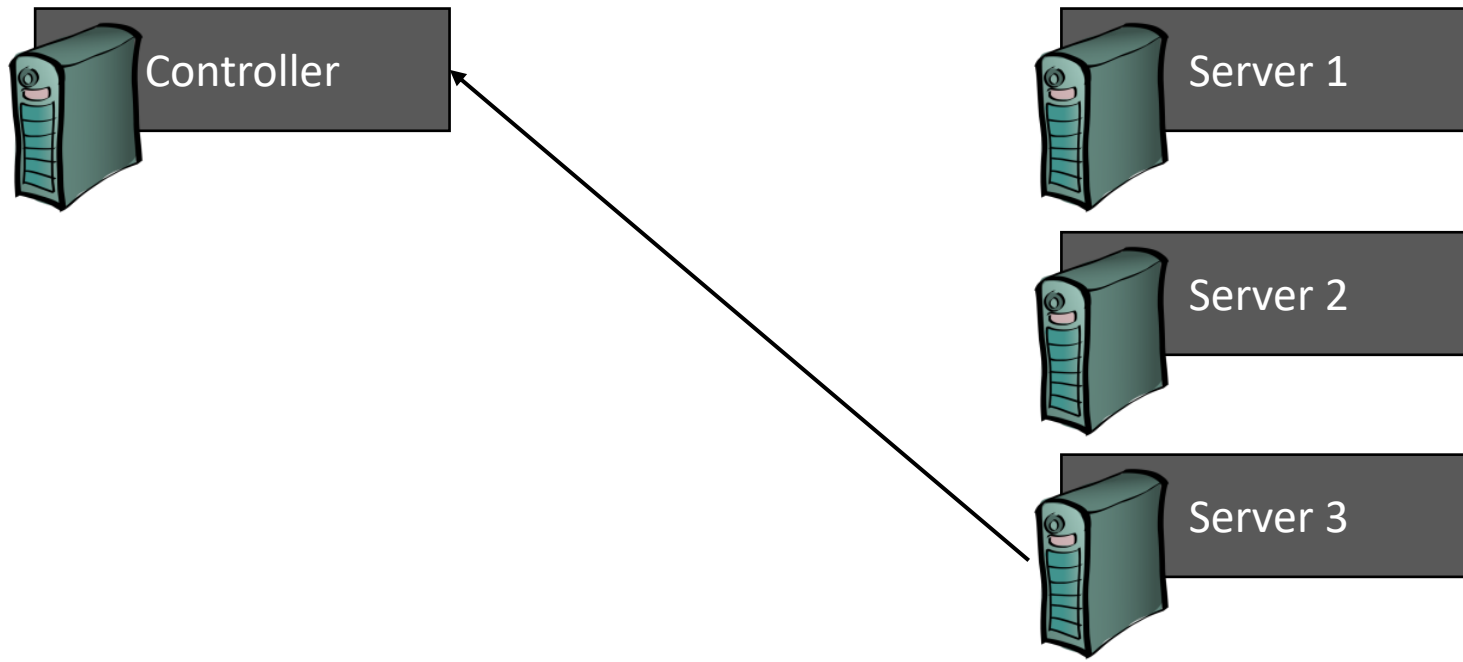
Push vs Pull

- Push → Central controller configures all devices
- Pull → Each device queries the central controller to update its config



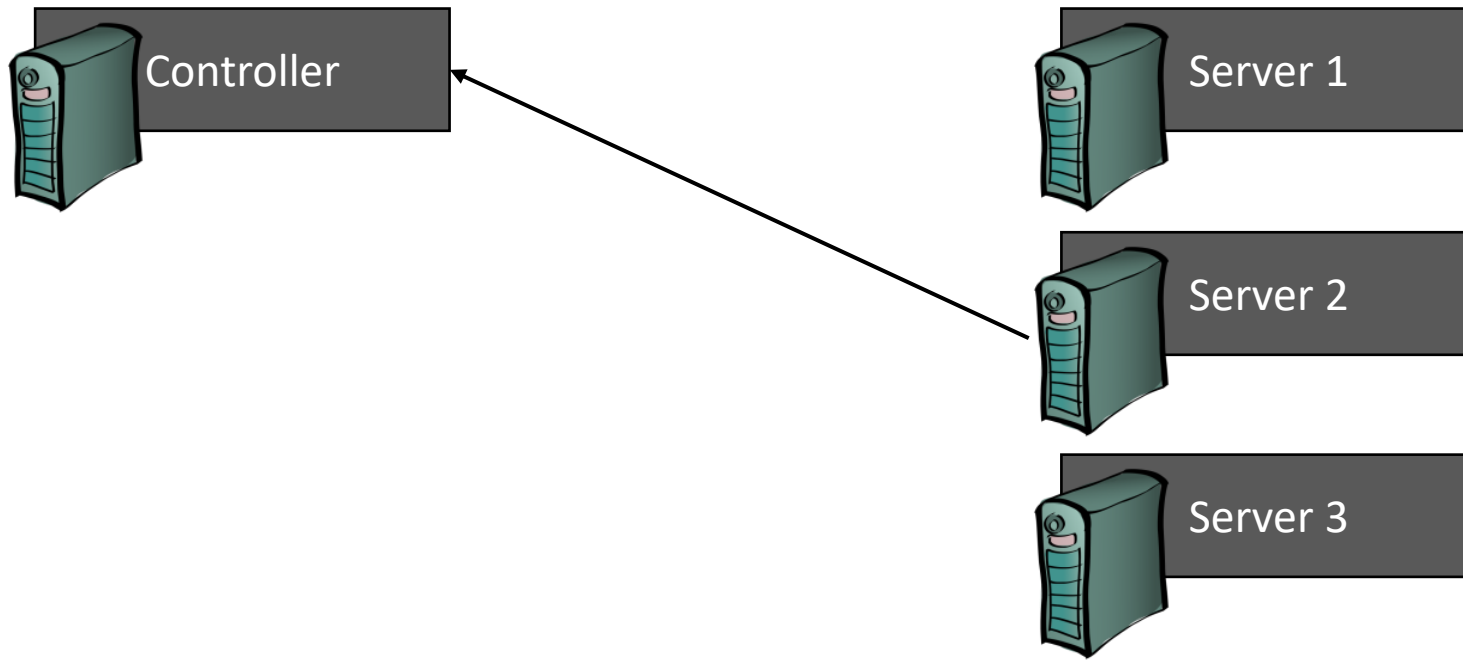
Push vs Pull

- Push → Central controller configures all devices
- Pull → Each device queries the central controller to update its config



Push vs Pull

- Push → Central controller configures all devices
- Pull → Each device queries the central controller to update its config



Push vs Pull

- Push → Central controller configures all devices
 - Requires central node to know about all machines
- Pull → Each device queries the central controller to update its config
 - Requires an agent to run on each device