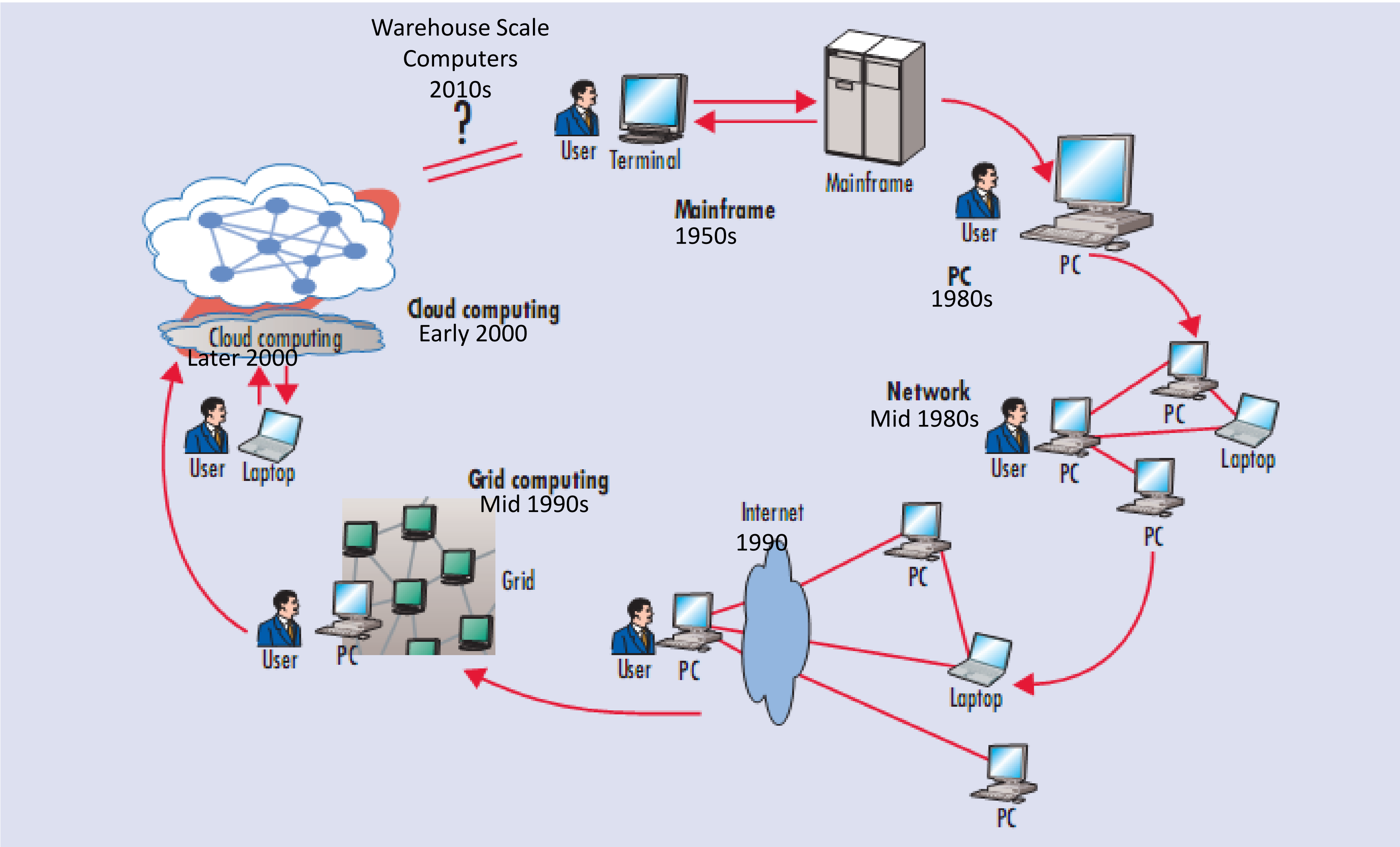
Evolution of Virtual Machines

Topic coverage

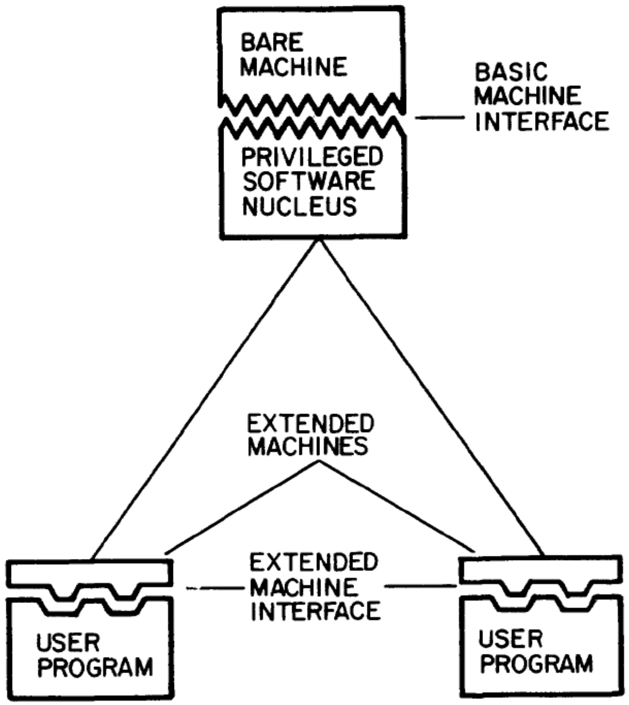
- Evolution of computing practices
- Why Virtual Machines?
- Motivation for Using Virtualization
- Data Center evolution
- Current day motivations for Virtual Machines.

Evolution of Computing Practices

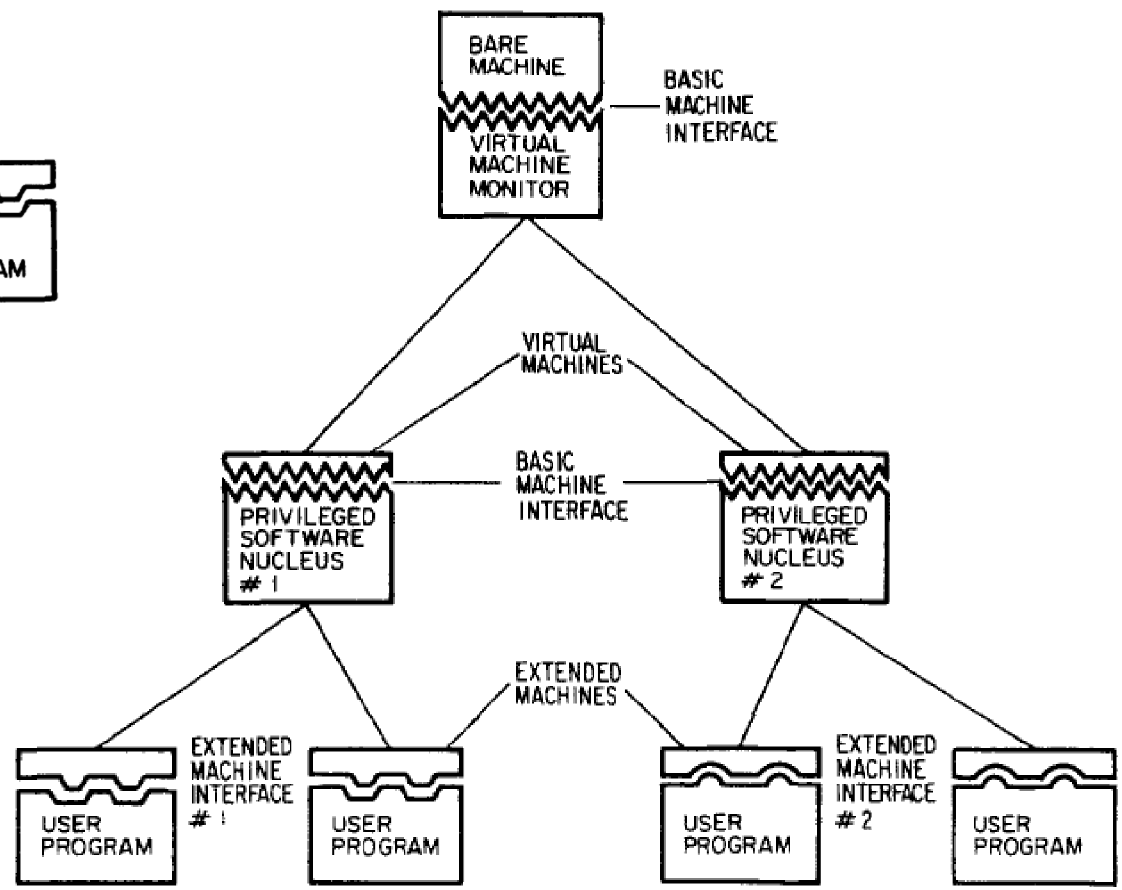


Why Virtual Machines?

- In the mainframe era (1960 – early 1980s)
  - Computer architecture change to support multi-user, multi-programming and introduction of I/O processors lead to dual-privilege processor architectures.
  - Introduced better system utilization but brought in more complexities associated with testing and development of new systems and system software.



BARE MACHINE
BASIC MACHINE INTERFACE
PRIVILEGED SOFTWARE NUCLEUS
EXTENDED MACHINES
EXTENDED MACHINE INTERFACE
USER PROGRAM
USER PROGRAM



BARE MACHINE
BASIC MACHINE INTERFACE
VIRTUAL MACHINE MONITOR
VIRTUAL MACHINES
BASIC MACHINE INTERFACE
PRIVILEGED SOFTWARE NUCLEUS # 1
PRIVILEGED SOFTWARE NUCLEUS # 2
EXTENDED MACHINES
EXTENDED MACHINE INTERFACE # 1
EXTENDED MACHINE INTERFACE # 2
USER PROGRAM
USER PROGRAM
USER PROGRAM
USER PROGRAM

Idea of the Privileged Software

- The privileged software of the OS-kernel was built to manage the system resources.
- The idea of multi-programming and multi-user is the functionality provided by the Privileged software.
- System hardware has sufficient constructs to support this idea of sharing through the Privileged software.
- Most of these sharing constructs happened to evolve around the idea of time-multiplexing of a resource.
  - Computing machinery was expensive so important to keep it busy as much as possible!

Premise for System Design

- Till year 2000 most system designs adopted approaches where single OS kernel is used for managing the resources.
- The applications are built to the extended machine functionality rather than the actual machine!

Era of PCs and Network of Computers

- Dual state architectures sufficed for PCs and later for applications built on Network of computers and subsequently for Distributed systems and Grid computing model.
- Many applications got built using the client-server or distributed services model.
- In all such scenarios, independent systems with their own privileged software topped with necessary application runtime software for distributed systems was sufficient to provide the desired functionality.

Concerns of Server Sprawl

- As hardware became cheaper many applications got built with their own hardware and associated software tiers.
- This model ensured applications delivering required performance.
- Increase in throughput or availability was mostly handled using isolated duplicate or redundant servers.
- With the advent of WWW this practise exploded into server sprawl!

Symptoms of Server Sprawl

- Enterprise server utilizations below 10%!
- Huge IT Capex but the inability to host newer applications due to mismatch of software requirements!
- Applications-Platform coupling led to unwarranted dependencies between unrelated applications!
  - Side-effects: failure of one application causes outages on the other!
- Ever increasing Opex bills!
- Precursor to Green computing!

Re-emergence of Virtualization

- Improve server utilization with application isolation.
- Enables co-hosting of multiple applications on single platform, each with their independent software and runtime environment – genesis for server consolidation.
- Offers platform independence by way of Virtual Machine encapsulation
  - Enables application scalability with varying workload demands
  - Improved application availability by way of isolation and elastic scaling capability
  - Faster provisioning for newer applications

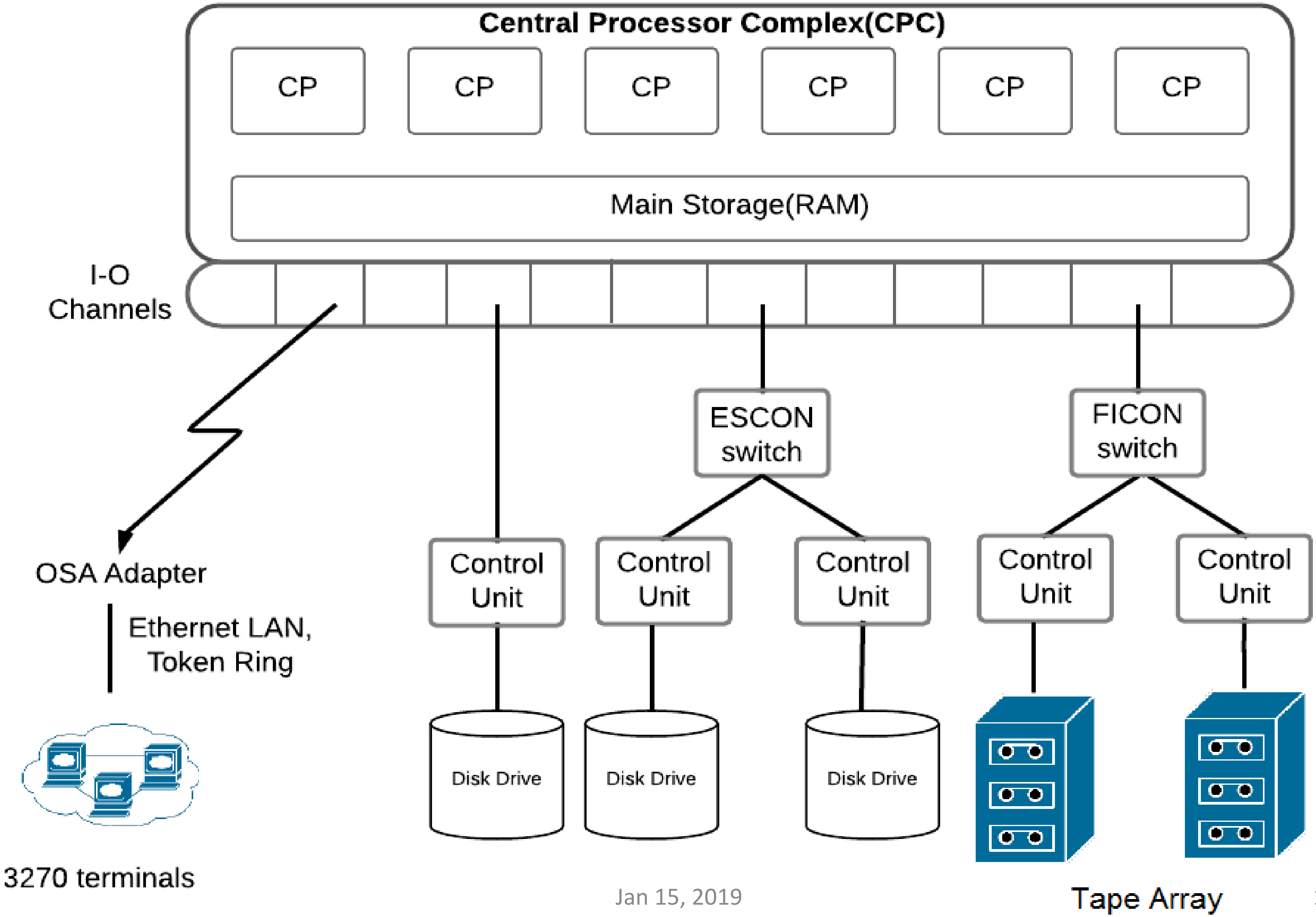
Cloud Setups

- Re-emergence of virtualization, service oriented architectures, the world wide web (Internet?) and utility computing paradigm enabled the realization of Cloud setups and cloud computing.
- Grid computing ushered in loose coupling of many, heterogeneous distributed systems through middleware (resource aggregation) for large-scale scientific computations with access from anywhere.
- Enterprises realized the benefit of segregation using virtualization with middleware for seamless use of computing infrastructure.

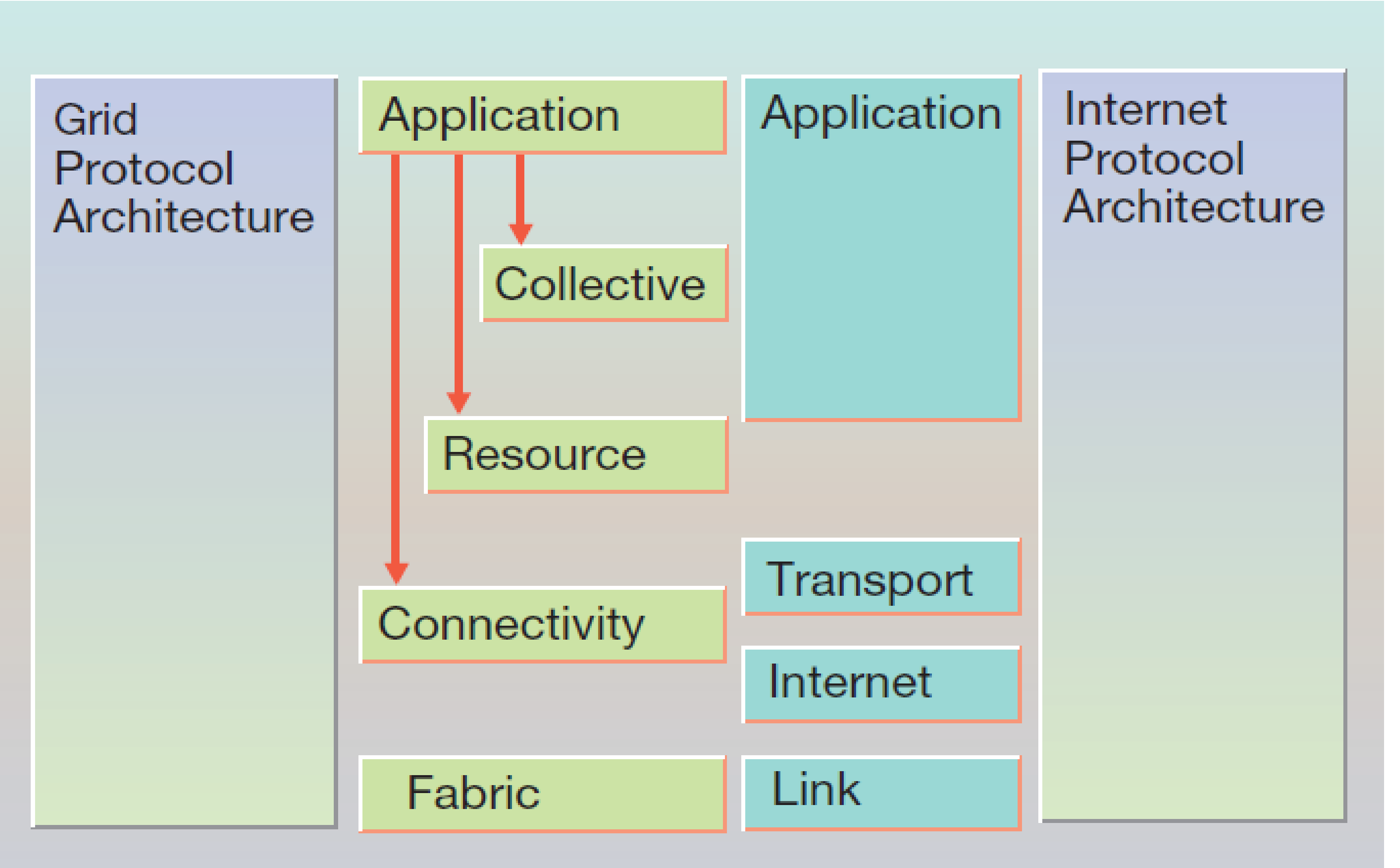
Data Center Evolution

- Mainframes:
  - Expensive hardware; virtualization enabled multi-user and multi-tenanted OS for system development
- Distributed Systems and Grid Computing: (Scientific workloads)
  - Multiple independent systems for increased throughput or availability of applications.
  - Limiting processor speeds forced users to use many cores for improved application performance using parallel programming.
- Cloud Data centres: (Enterprise workloads)
  - Server consolidation through virtualization yields improved server utilization and reduced power and real-estate footprints.
- Hyperconverged data centres:
  - Heterogeneous platforms with server virtualization results in resource fragmentation
  - Software defined data centers with commensurate hardware interconnects enables flexible hardware compositions for better utilization at data centers!

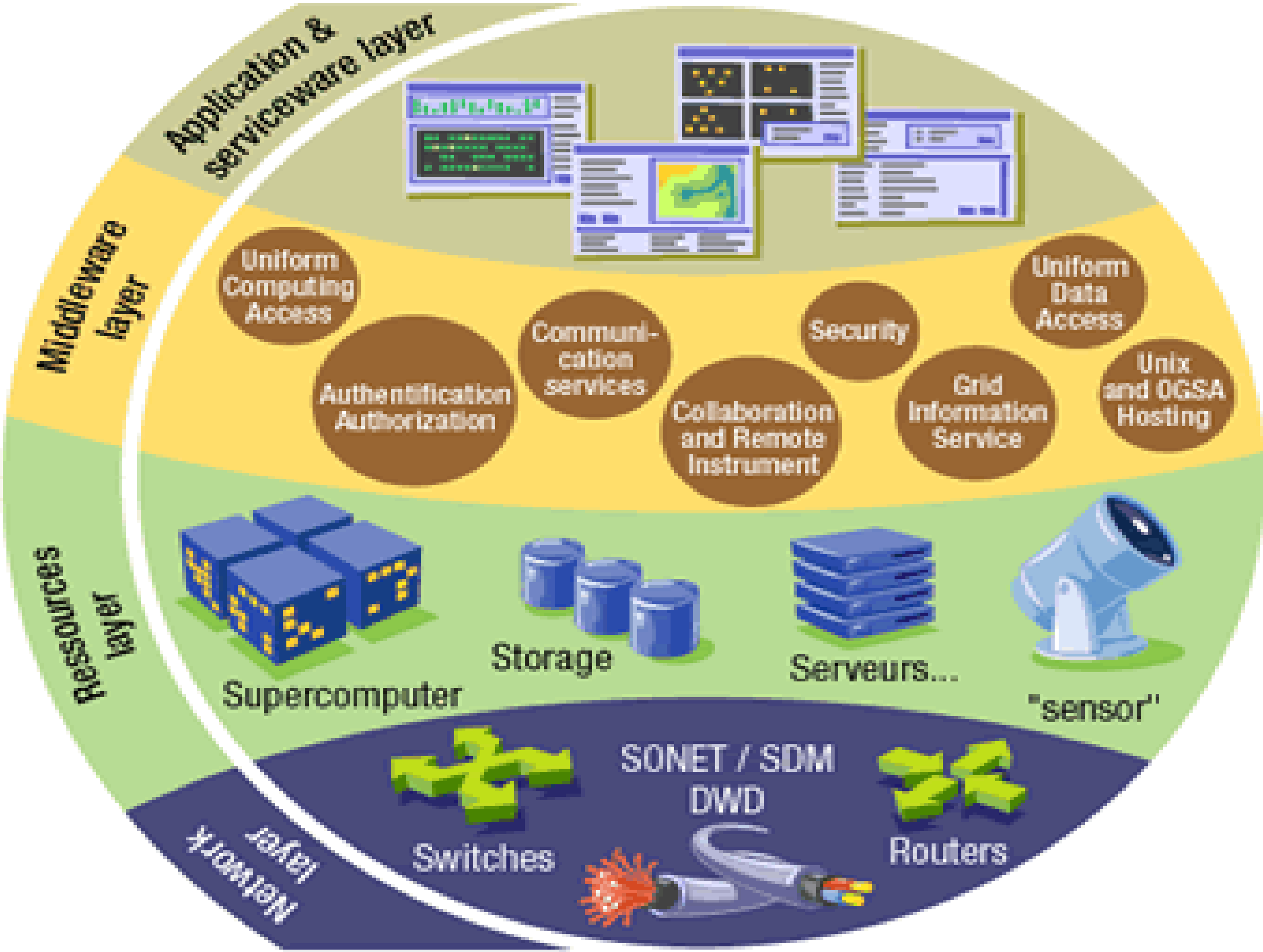
Mainframes


Central Processor Complex(CPC)
CP
CP
CP
CP
CP
CP
Main Storage(RAM)
I-O Channels
ESCON switch
FICON switch
OSA Adapter
Ethernet LAN, Token Ring
Control Unit
Control Unit
Control Unit
Control Unit
Control Unit
Disk Drive
Disk Drive
Disk Drive
3270 terminals
Tape Array

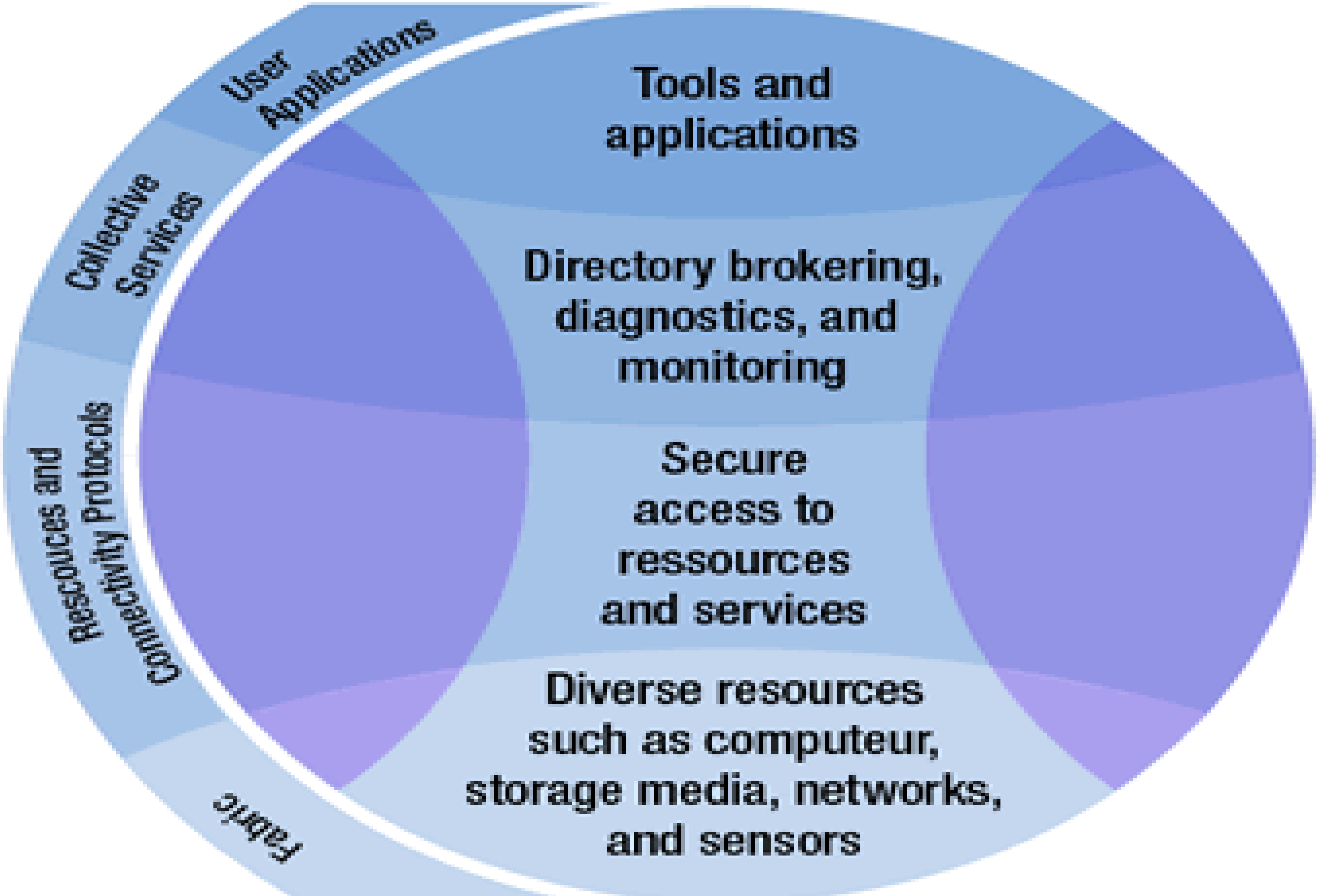
Grid Computing Architecture



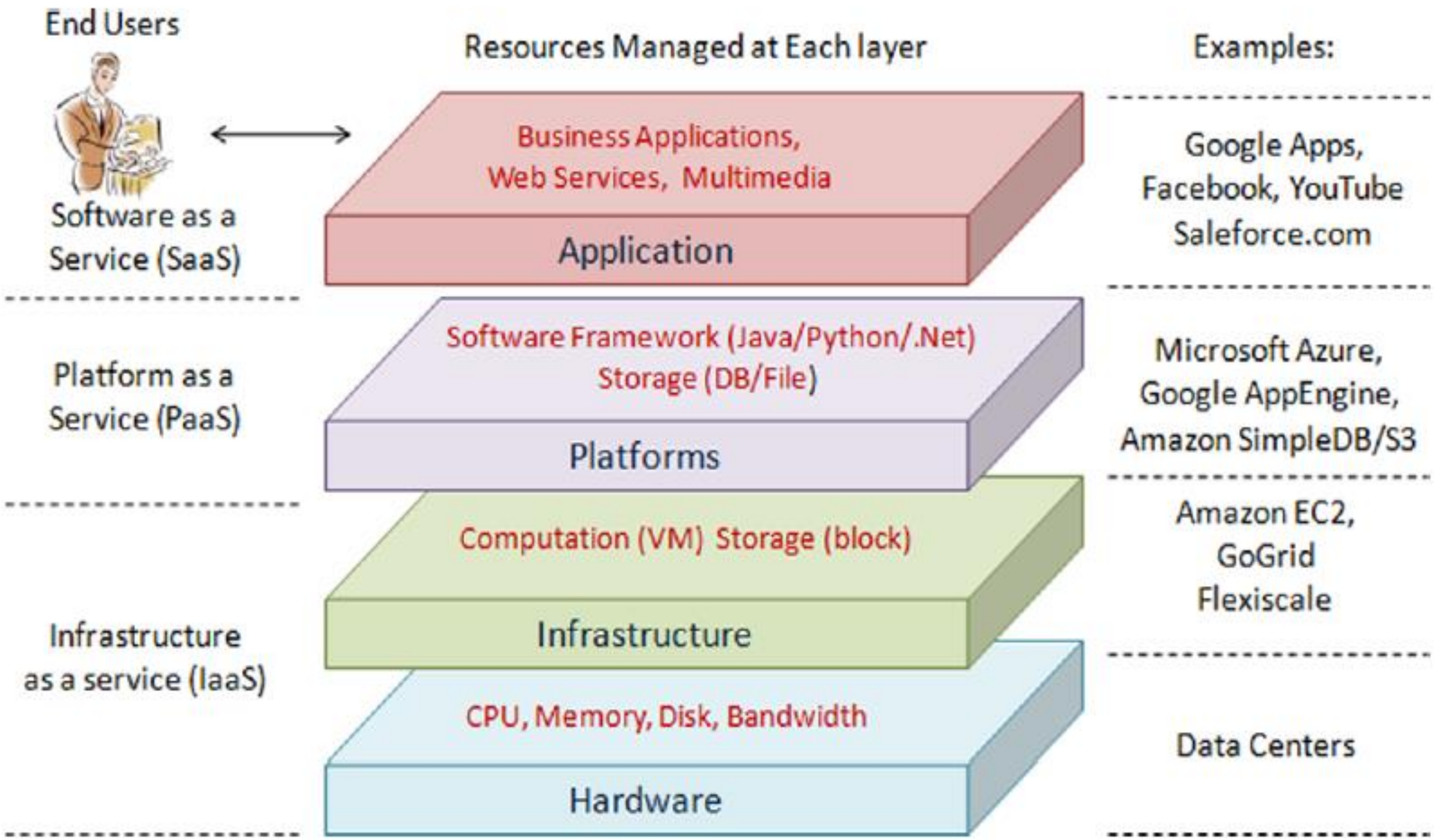
Grid computing Logical Layers



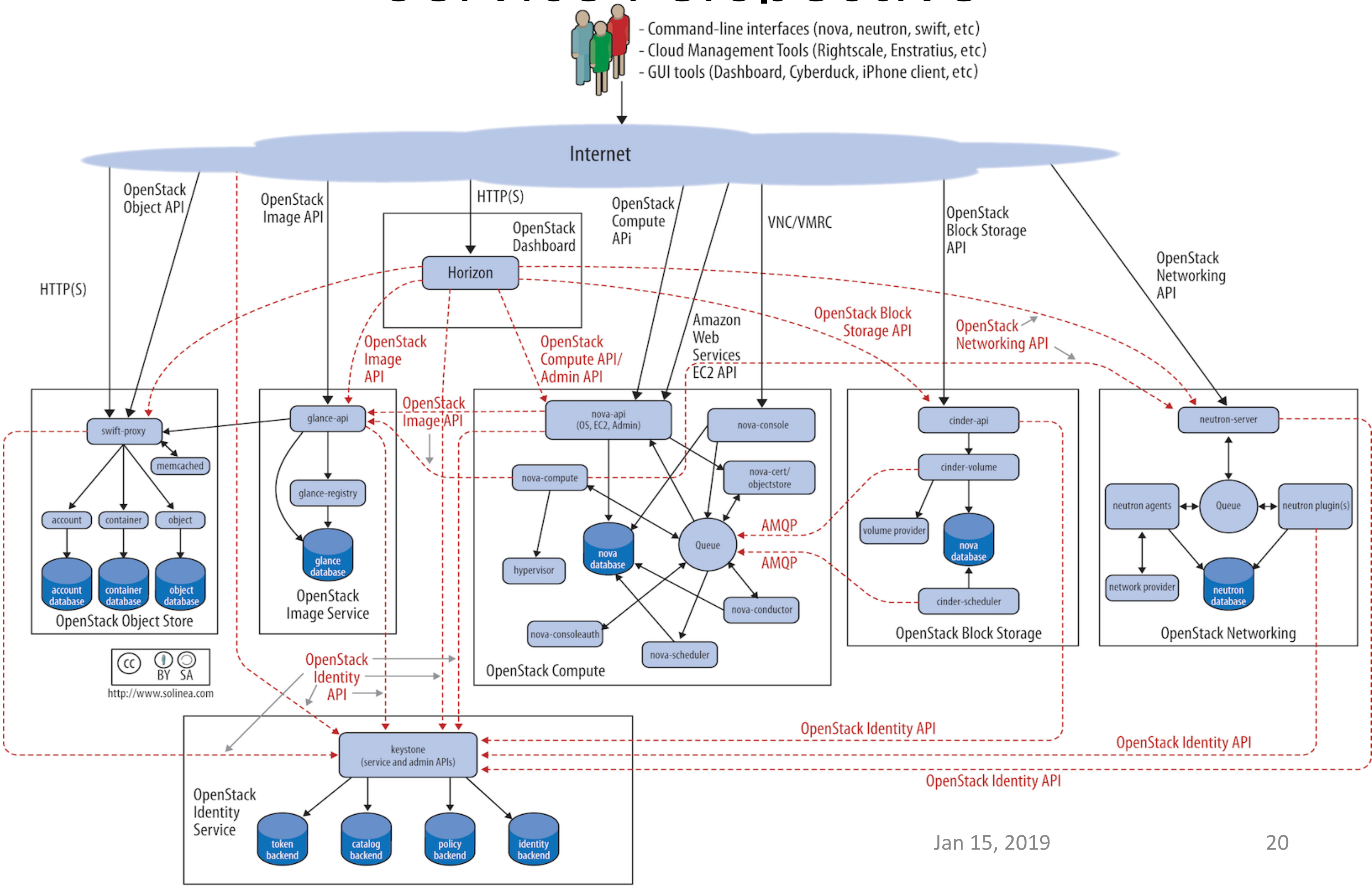
Grid Computing Service Layers



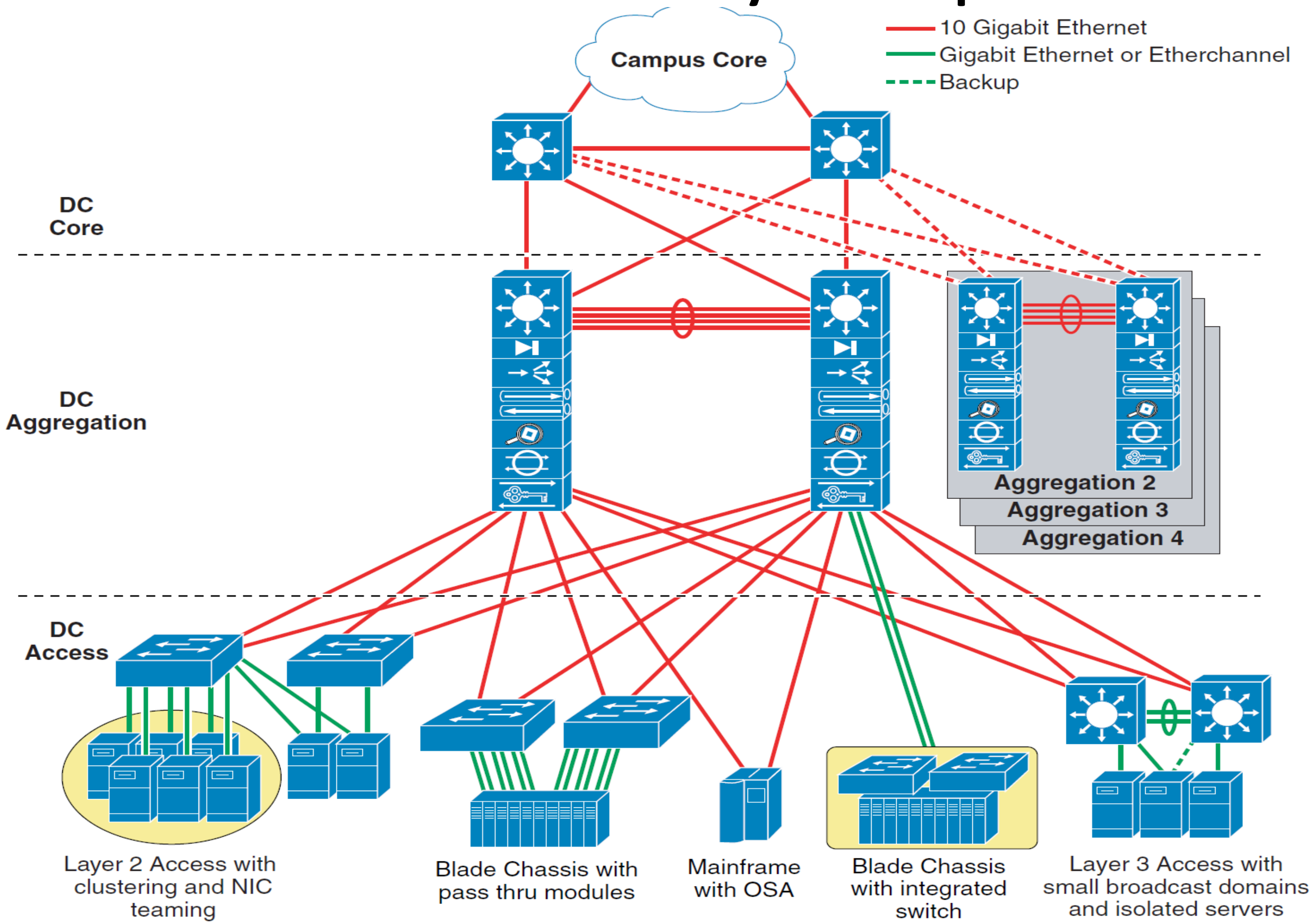
Cloud Computing Architecture Logical and Service Layers



OpenStack Cloud Architecture Service Perspective



Data Centre Architecture
Fabric Connectivity Perspective



Source: https://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Data_Center/DC_Infra2_5/DCI_SRND_2_5a_book.pdf

Hyper-Converged Data Centres

- Integrated systems:
  - Initial Cloud setups started with virtualized servers and storage integrated using cloud stack.
  - Simple conglomerations of existing hardware and software with SAN based storage access
  - Vendor lock-in with server and storage OEMs
- Converged Infrastructure:
  - Server and storage components converged to a single appliance (VM)
  - Unified and simplified management and faster deployment
  - Resource ratios (cpu:storage:network) fixed and hence inflexible and have performance utilization conflicts.
  - Legacy applications still need to be re-provisioned or migrated to cloud infrastructure
- Hyper-converged Infrastructure:
  - Consolidation of required functionality into a single infrastructure stack implemented on simple, efficient and elastic resource pool
  - Software Defined Data Centers (SDDCs) enable the idea of convergence of hardware along-with functionalities like backup, replication, deduplication, elasticity, network gateways, high speed storage access through SSD cache and drive arrays, etc.

Convergence Characteristics

	Technical Attributes			Organizational Benefits	
	Data Efficiency	Single Shared Resource Pool	Global Management	TCO Improvements	Simplification
Convergence 1.0	☐	☐ Resource pooling limited to server layer	☐	☐	⊡ Some time to deployment and administrative gains
Convergence 2.0	☐	⊡ Limited to primary server and storage resources; other resources not included	☐	⊡ TCO gains primarily due to redunction of legacy gear; does not address backup, replication, and DR	⊡ Fewer products to manage
Convergence 3.0	▣ Data architecture begins with one-time deduplication, compression, and optimization of data	▣ All data center resources are brought into the resource stack	▣ Complete management of all infrastructure resources and virtual machines; single point of administration	▣ Major TCO gains through reduction of hardware resources, streamlined operations, and automation	▣ Reduces hardware littered across data centers, eases management, VM-centricity

☐ ***Not supported*** ⊡ ***Partially Supported*** ▣ ***Fully Supported***

Current-day Motivations for using Virtualization

1. RTVirt: Virtualization brings the capability of agile adaptivity for deploying applications, however due to multiple layers of indirection, poses several challenges for real-time constrained applications. RTVirt is a paper in the direction of using co-ordinated multi-layer scheduling algorithms to deliver real-time response at the application layer. The results show that it can meet application deadlines (99%) or tail latency requirements (99.9th percentile) nearly perfectly; it can handle large numbers of applications and dynamic changes in their timeliness requirements; and it substantially outperforms the existing solutions in both timeliness and resource utilization.

Ming Zhao and Jorge Cabrera. 2018. RTVirt: enabling time-sensitive computing on virtualized systems through cross-layer CPU scheduling. In Proceedings of the Thirteenth EuroSys Conference (EuroSys '18). ACM, New York, NY, USA, Article 27, 13 pages. DOI: https://doi.org/10.1145/3190508.3190527

Current-day Motivations for using Virtualization

2. NFV: Network Function virtualization (NFV) is a technique to build and deploy network functions on virtual machine instances. NFVs decouple network services from customised proprietary installations to commodity based VMs and enable network services to scale by demand. This study proposes using NFV technology and found that the cost was reduced by 24% compared to existing network equipment.

Jun-Ho Huh, Server Operation and Virtualization to Save Energy and Cost in Future Sustainable Computing, Sustainability 2018, 10(6), 1919; doi:10.3390/su10061919

Current-day Motivations for using Virtualization

3. Lightweight Virtualization (LV) for IoT Edge Computing: Edge computing deals with building viable software constructs to handle IoT data near source. This paper discusses and compares the applicability of two LV technologies, containers and unikernels, as platforms for enabling the scalability, security, and manageability required by such pervasive applications.

R. Morabito, V. Cozzolino, A. Y. Ding, N. Beijar and J. Ott, "Consolidate IoT Edge Computing with Lightweight Virtualization," in IEEE Network, vol. 32, no. 1, pp. 102-111, Jan.-Feb. 2018.

doi: 10.1109/MNET.2018.1700175

Current-day Motivations for using Virtualization

4. PVFlow: Software Defined Networks (SDNs) bring the fluidity required to handle dynamic network configurations as demanded by the cloud setups. Coupled with network virtualization, SDNs using flow table virtualization can realize virtual SDNs (vSDNs) for enhanced programmability, adaptivity and cost-effectiveness.

S. Li, K. Han, H. Huang and Z. Zhu, "PVFlow: Flow-Table Virtualization in POF-based vSDN Hypervisor (PVX)," *2018 International Conference on Computing, Networking and Communications (ICNC)*, Maui, HI, 2018, pp. 861-865.
doi: 10.1109/ICCNC.2018.8390375

Current-day Motivations for using Virtualization

5. Unikernels: Unikernels are single-purpose appliances that are compile-time specialised into standalone kernels, and sealed against modification when deployed to a cloud platform. In return they offer significant reduction in image sizes, improved efficiency and security, and should reduce operational costs.

Anil Madhavapeddy, Richard Mortier, Charalampos Rotsos, David Scott, Balraj Singh, Thomas Gazagnaire, Steven Smith, Steven Hand, and Jon Crowcroft. 2013. Unikernels: library operating systems for the cloud. SIGARCH Comput. Archit. News 41, 1 (March 2013), 461-472. DOI: https://doi.org/10.1145/2490301.2451167

Current-day Motivations for using Virtualization

6. IncludeOS: IncludeOS, a single tasking library operating system for cloud services, written from scratch in C++. Cloud service that needs to be both elastic and resource efficient needs A) highly specialized components, and B) to run with minimal resource overhead. Classical general purpose operating systems designed for extensive hardware support are by design far from meeting these requirements.

A. Bratterud, A. Walla, H. Haugerud, P. E. Engelstad and K. Begnum, "IncludeOS: A Minimal, Resource Efficient Unikernel for Cloud Services," 2015 IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom), Vancouver, BC, 2015, pp. 250-257. doi: 10.1109/CloudCom.2015.89

Summary

- Evolution of computing practices
  - Usage Perspective
  - System Architecture perspective
- Why Virtual Machines?
- Motivation for Using Virtualization
- Data Center evolution
  - Why it is relevant to understand system virtualization
- Current day motivations for Virtual Machines

Questions?

Thankyou!