

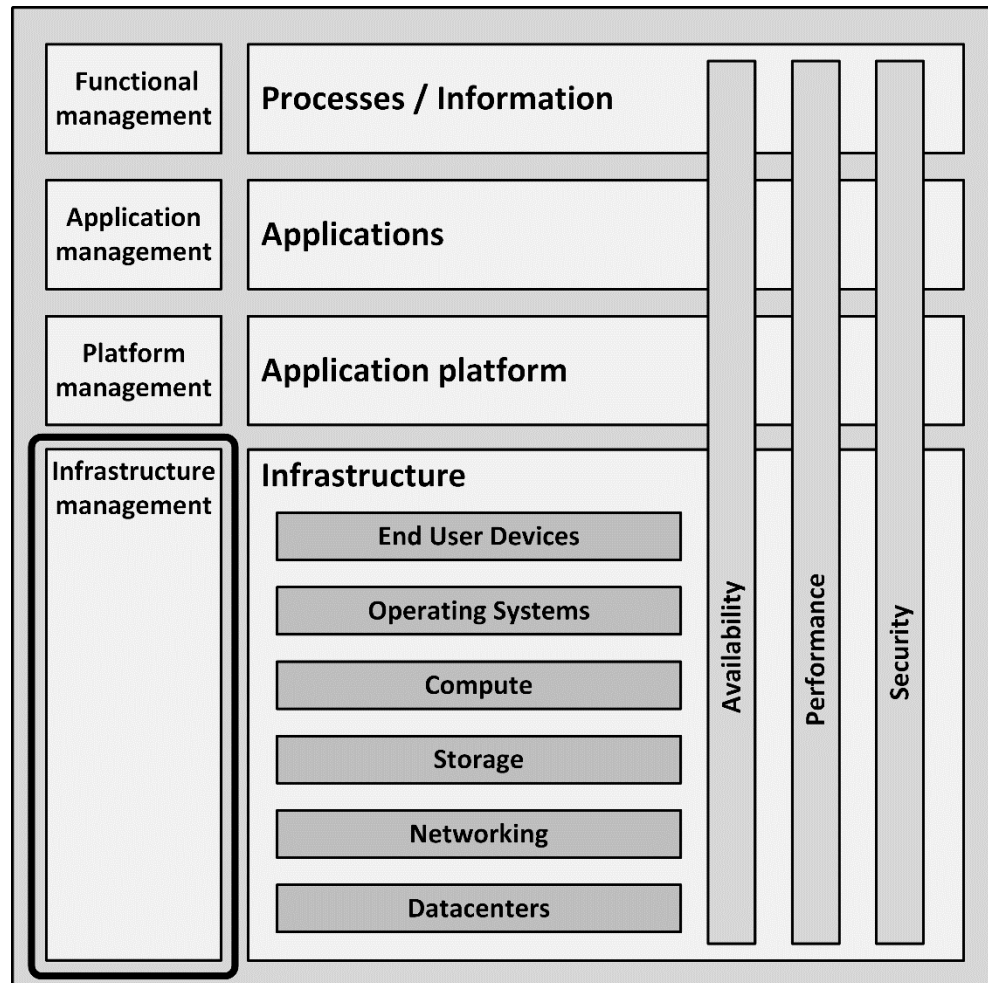
IT Infrastructure Architecture

Infrastructure Building Blocks
and Concepts

Hosting and deployment options

Introduction

- This part is about the systems management processes to determine the best deployment option for the infrastructure



Hosting options

- Hosting options define *where* infrastructure is deployed
- Deployment models define *how* infrastructure is deployed
- Infrastructure can be hosted:
 - On-premises
 - In a colocation
 - Outsourced

On-premises hosting

- Infrastructure components run on the premises of the organization using the infrastructure
 - In the datacenter of an existing building
 - In a dedicated, specially designed datacenter building
- The building must have:
 - Enough space
 - An uninterruptable power supply (UPS)
 - Options to install sufficient cooling
 - Fire prevention and detection
 - External redundant network capabilities with enough bandwidth
 - Sufficient floor loading capacity

On-premises hosting

- Drawbacks:
 - On-premises datacenters don't scale well, as they are embedded in existing (office) buildings
 - The organization must have enough knowledge and staff available to manage the datacenter

Colocation

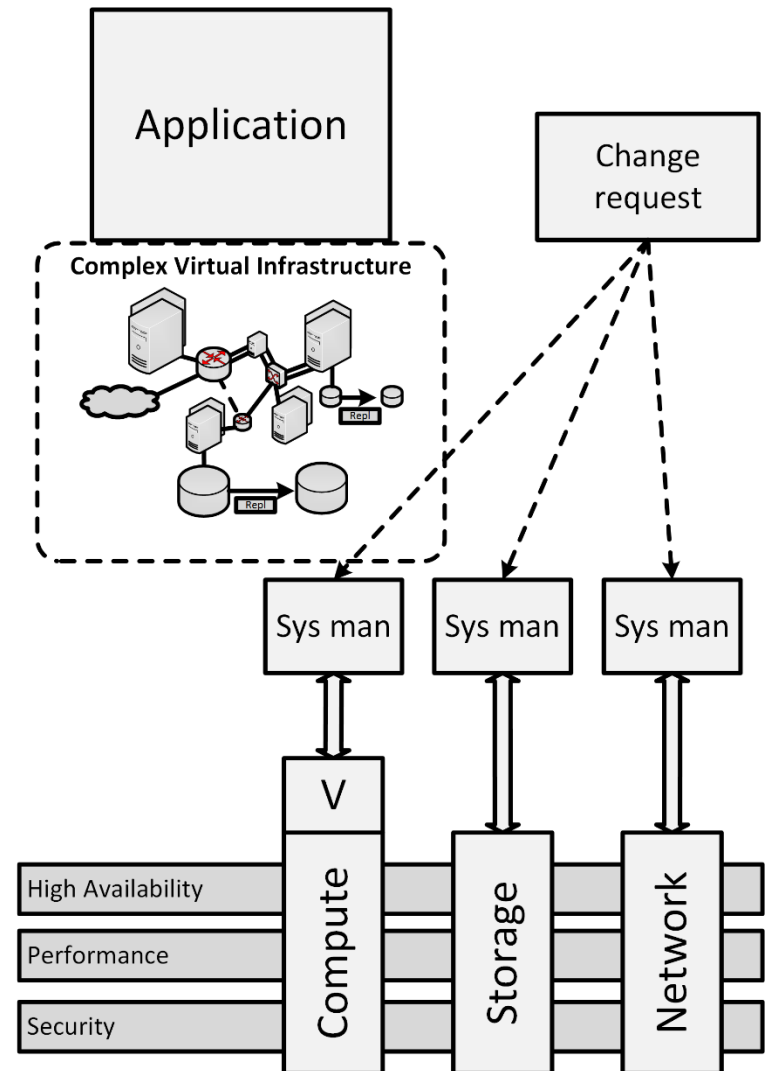
- A third party dedicated datacenter where racks, floor space, and network bandwidth can be rented
- Hosts and connects customer owned infrastructure components
- Provides:
 - Power
 - Cooling
 - Physical security
- Racks are empty – all infrastructure components must be provided and managed by the organization renting the colocation racks

Outsourcing

- Full infrastructure outsourcing is a subcontracting service in which some third-party purchases, deploys, hosts, and manages the infrastructure, and performs its lifecycle management
 - Managed using Service Level Agreements
 - Very rigid change management process
- Frees the organization from investing in hardware
 - Only operational cost
- The outsourcing organization must have:
 - A demand organization
 - A process to manage the outsourcing party
- The outsourcing organization can be freed from internal infrastructure systems managers

Enterprise infrastructure deployment

- A traditional enterprise infrastructure deployment
- Enterprise grade hardware delivers three infrastructure resources:
 - Compute
 - Storage
 - Network
- Provides:
 - High availability
 - High performance
 - Some security controls



Enterprise infrastructure deployment

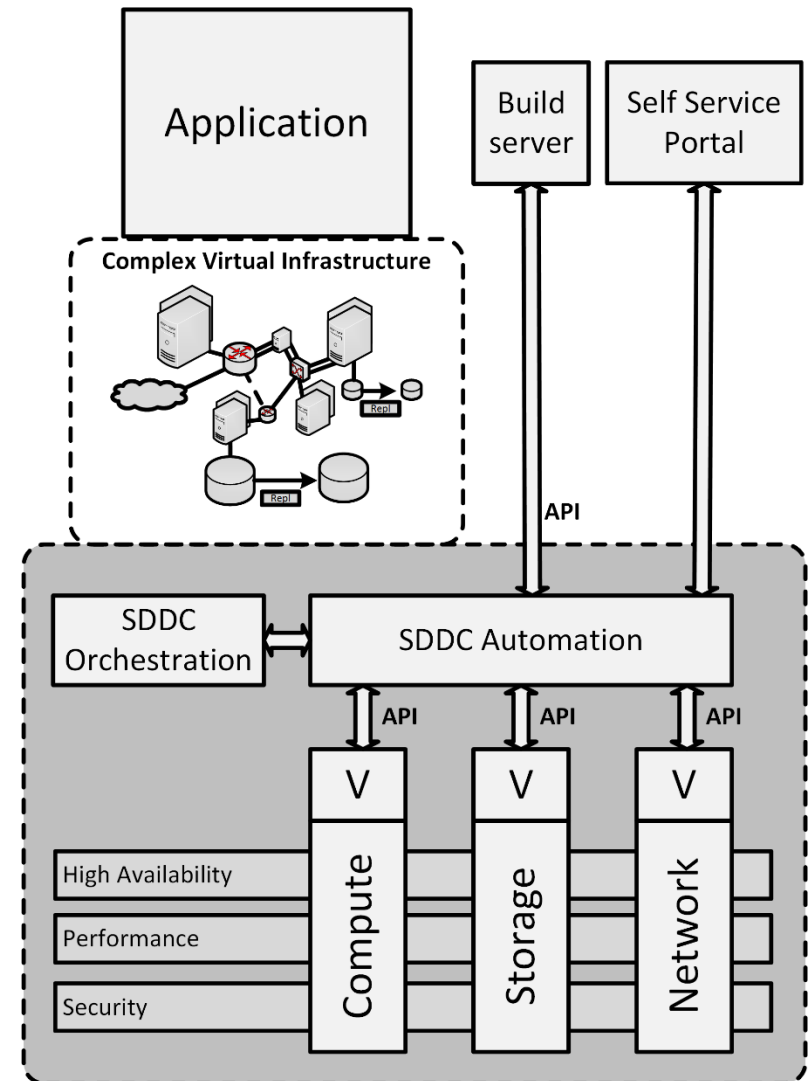
- Most enterprise infrastructure deployments implement compute virtualization to provide virtual machines
- Each resource is managed by a team of dedicated systems managers
- Changes are managed by a workflow based change process
 - Each systems manager is responsible to manually perform his part of the change

Enterprise infrastructure deployment

- Enterprise infrastructures can deliver complex virtual infrastructures to applications
- The virtual infrastructure is designed to meet the requirements of applications
- Applications depend on the high availability, performance and security controls delivered by the infrastructure

Software-defined datacenter - SDDC

- A software-defined datacenter (SDDC) is an architecture in which all infrastructure resources – compute, storage and networking – are virtualized, and can be configured using software APIs
- An SDDC is an extension of an enterprise infrastructure
 - All resources are virtualized and managed by SDDC automation and orchestration software



Software-defined datacenter - SDDC

- An SDDC is characterized by:
 - Automation
 - Orchestration
 - Abstraction of resources into software and code
- Changes are managed by an automated workflow
- An orchestrated change can lead to a number of automated changes in various resources

Software-defined datacenter - SDDC

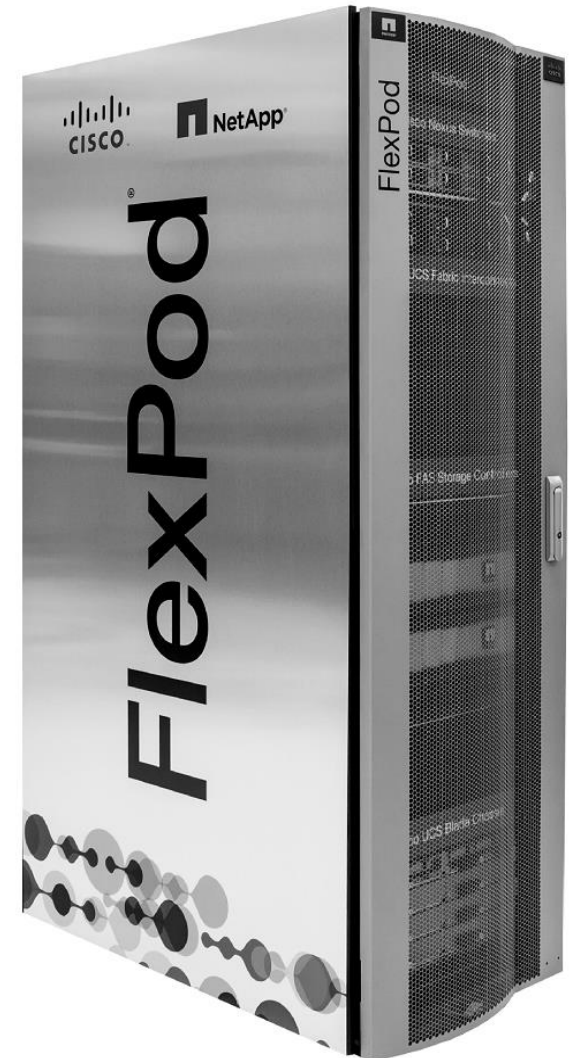
- An SDDC is the foundation for cloud computing
- Developers, DevOps teams and systems managers can create and deploy new infrastructures using:
 - A manual self-service portal
 - A combination of a build server and APIs
- Allows the user to request:
 - Desired infrastructure components
 - Their sizing to meet performance demands
 - Their required availability

Software-defined datacenter - SDDC

- SDDC software provides tools for:
 - Costing
 - Logging
 - Reporting
 - Scaling (up and down)
 - Decommissioning of the infrastructure resources
- Examples of SDDC automation and orchestration products:
 - OpenStack Horizon
 - IBM Cloud Orchestrator
 - VMware vRealize

Converged Infrastructure

- Compute, storage, and network components are designed, assembled, and delivered by one vendor and managed as one system, typically deployed in one or more racks



Hyperconverged Infrastructure

- Compute, storage, and network components are implemented within a single server node
- A hyperconverged infrastructure comprises:
 - A large number of identical physical servers from one vendor
 - Direct attached storage in the server
 - Special software that manages all servers, storage, and networks as one cluster running virtual machines



Hyperconverged Infrastructure

- The technology is easy to expand on-demand, by adding nodes to the hyperconverged cluster
- Hyperconverged systems are an ideal candidate for deploying VDI environments
 - Storage is close to compute (in the same node)
 - Scales well with the rise in the number of users (add more nodes)

(Hyper)converged Infrastructure

- Advantages:
 - Only one vendor
 - All updates for compute, storage and networking in one service pack
 - Easy scalable
- Disadvantages:
 - Vendor lock-in – the solution is only beneficial if all infrastructure is from the same vendor
 - Scaling can only be done in fixed building blocks
 - If more storage is needed, compute must also be purchased

Cloud computing

- Cloud computing is one of the most important paradigm shifts in computing in recent years
- Definition NIST:

*Cloud computing is a model for enabling ubiquitous, convenient, **on-demand network access** to a **shared pool** of configurable computing resources that can be **rapidly provisioned** and released with **minimal management effort** or service provider interaction*

- Cloud computing is an outsourcing model, in which IT services are provided and paid based on actual on-demand use

Cloud characteristics

- **On demand self-service** – End users can configure, deploy, start and stop systems or applications on demand
- **Rapid elasticity** – A cloud is able to quickly scale-up and scale-down resources, leading to elasticity of resources
- **Resource pooling** – Instead of providing each application with a fixed amount of processing power and storage, cloud computing provides applications with resources from a shared pool
- **Measured service** – The actual resource usage is measured and billed. There are no capital expenses, only operational expenses
- **Broad network access** – Capabilities are available over the network and accessed through standard mechanisms

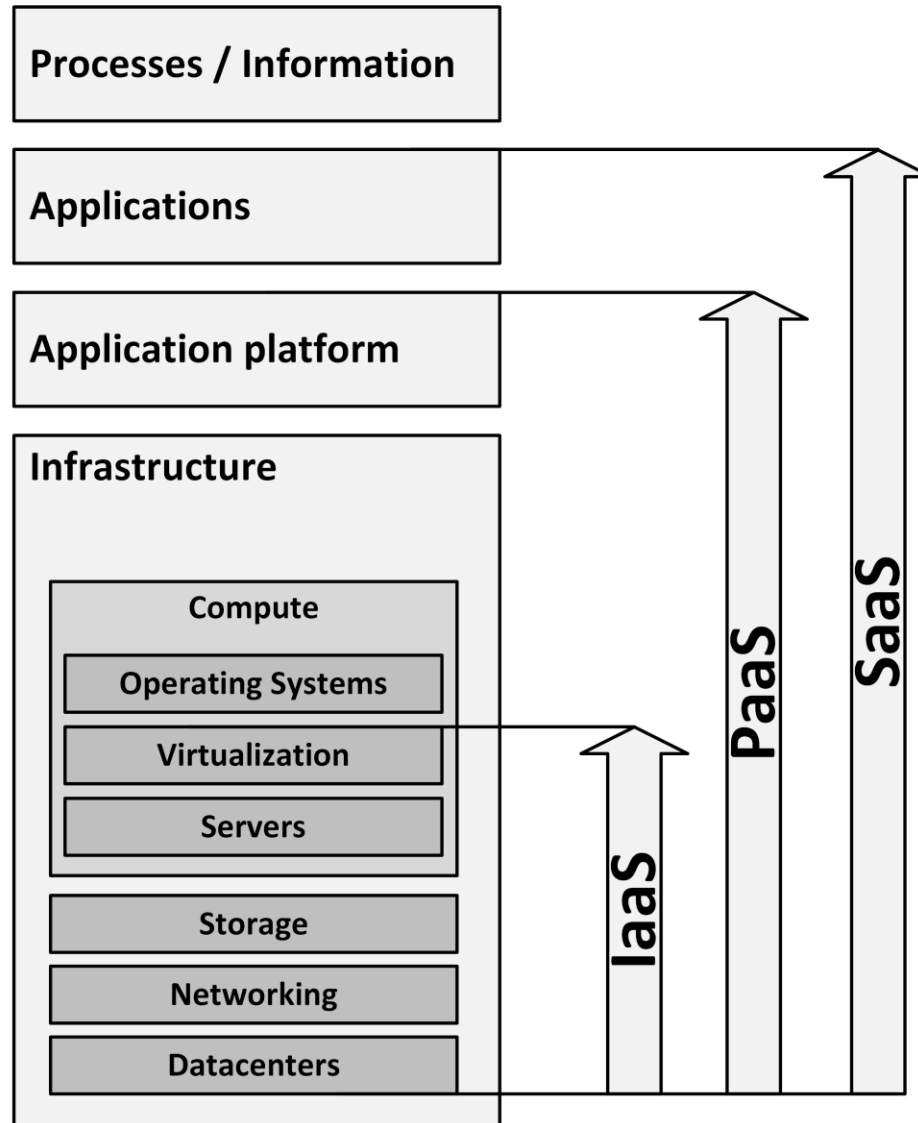
Cloud deployment models

- A **public cloud** is accessible through the internet, and available to the general public
- A **private cloud** is operated solely for a single organization, whether managed internally or by a third-party, and hosted either on premises or external
- A **community cloud** is much like a private cloud, but shared with a community of organizations that have shared concerns
- In a **hybrid cloud** deployment, a service or application is provided by a combination of a public cloud, and a community cloud and/or a private cloud

Cloud service models

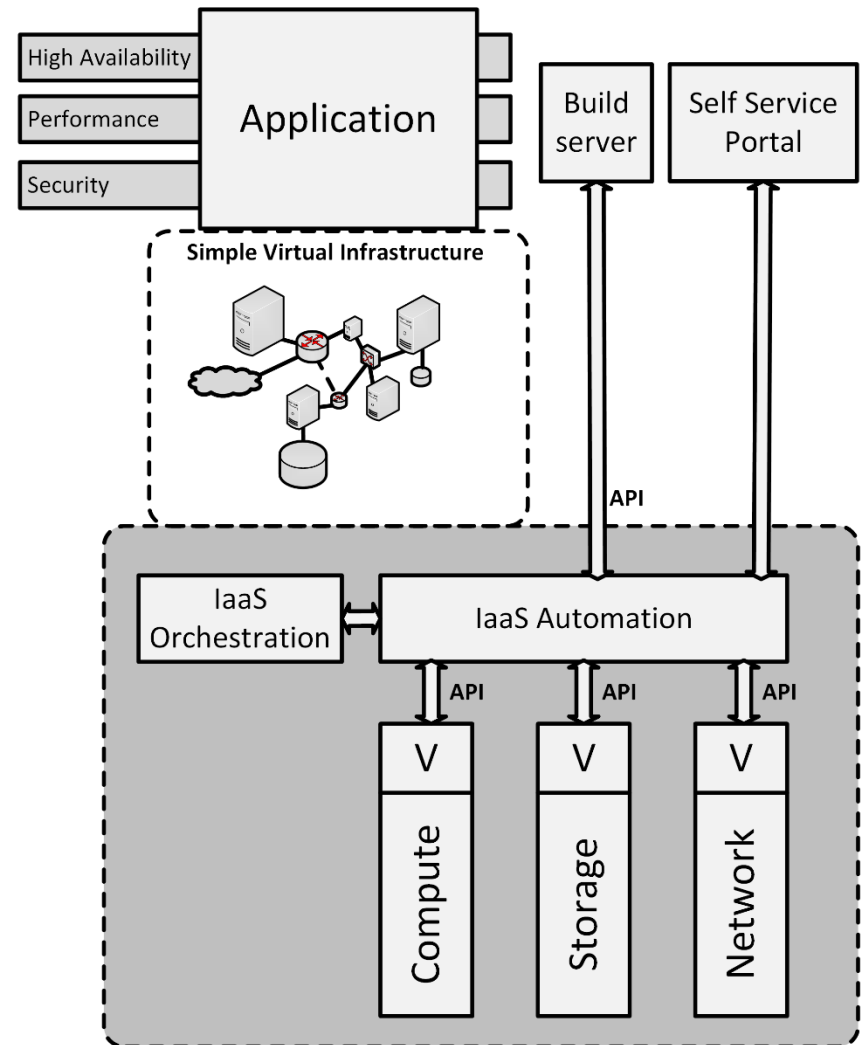
- **Software-as-a-Service (SaaS)** delivers full applications that need little or no configuration
 - Microsoft Office365, LinkedIn, Facebook, Twitter, and Salesforce.com
- **Platform-as-a-Service (PaaS)** delivers a scalable, high available, open programming platform that can be used by developers to build bespoke applications that run on the PaaS platform
 - Microsoft Azure Cloud Service and Google App Engine
- **Infrastructure-as-a-Service (IaaS)** delivers virtual machines, networking, and storage
 - Amazon Elastic Cloud (EC2 and S3) and Microsoft Azure IaaS

Cloud service models



Infrastructure as a Service (IaaS)

- Provides:
 - Virtual machines
 - Virtualized storage
 - Virtualized networking
 - Systems management tools to manage them



Infrastructure as a Service (IaaS)

- Failure of the infrastructure is acceptable:
 - Based on cheap commodity white label hardware
 - The philosophy is to keep the cost down by allowing the hardware to fail every now and then
 - Failed components are either replaced or simply removed from the pool of available resources

Infrastructure as a Service (IaaS)

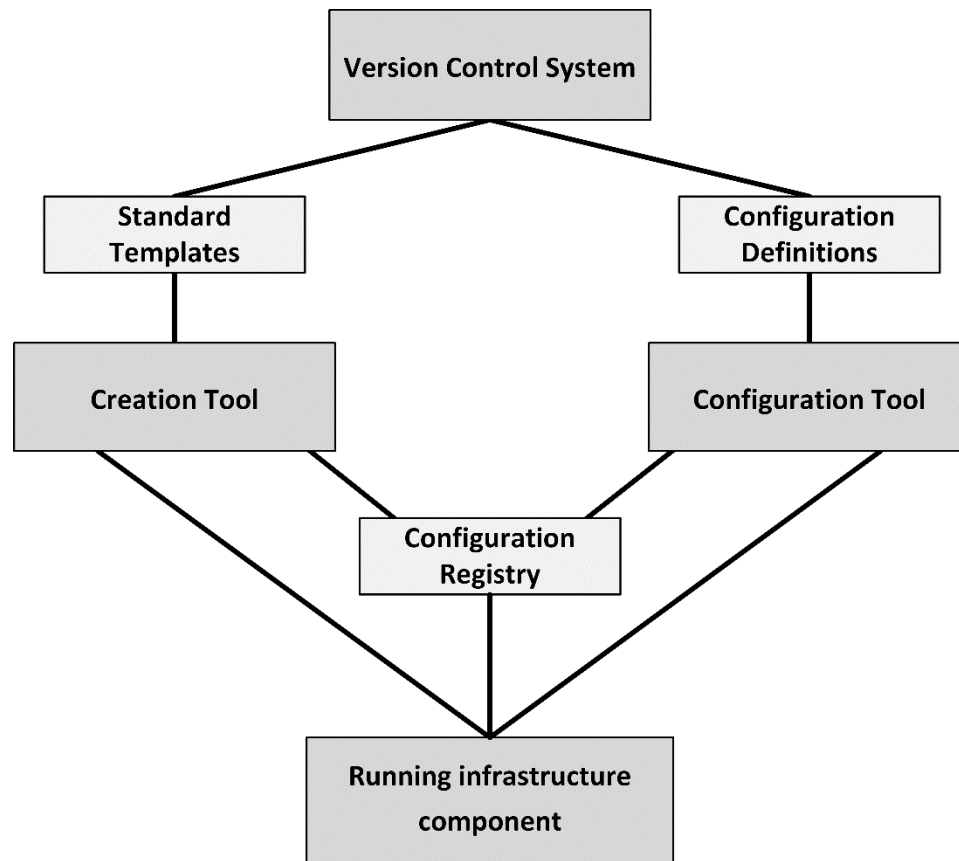
- IaaS provides simple, highly standardized building blocks to applications
 - Does not provide high availability, guaranteed performance or extensive security controls
 - Applications should be:
 - Robust to allow for failing hardware
 - Horizontally scalable to increase performance

Infrastructure as a Service (IaaS)

- To use IaaS, users must create and start a new server, and then install an operating system and their applications
- The cloud provider only provides basic services, like billing and monitoring
- The user is responsible for patching and maintaining the operating systems and application software
- Not all operating systems and applications can be used in an IaaS cloud
 - Many software licenses prohibit the use of a fully scalable, virtual environment like IaaS

Infrastructure as code

- In infrastructure as code, servers, storage, and networks can be created and configured automatically
- Tools to implement infrastructure as code include:
 - Puppet
 - Chef
 - Ansible
 - SaltStack
 - Terraform



Infrastructure as code

- Infrastructure as code ensures all infrastructure components that should be equal, are equal.
 - Standard templates are defined that describe the basic setup of infrastructure components
 - Configurations of infrastructure components are defined in configuration definitions
 - New instances of infrastructure components can be created automatically by a creation tool
 - Using the standard templates
 - Leads to a running, unconfigured infrastructure component

Infrastructure as code

- After an infrastructure component is created, the configuration tool automatically configures it
 - Based on the configuration definitions
 - Leads to a running, configured infrastructure component
- When the new infrastructure component is created and configured, its properties are automatically stored in the configuration registry
 - DNS name
 - Is the server is part of a load balancer pool?

Infrastructure as code

- The configuration registry allows running instances of infrastructure to recognize and find each other
 - It ensures all needed components are running
- Configuration definition files and standard templates are kept in a version control system
 - Enables roll backs and rolling upgrades