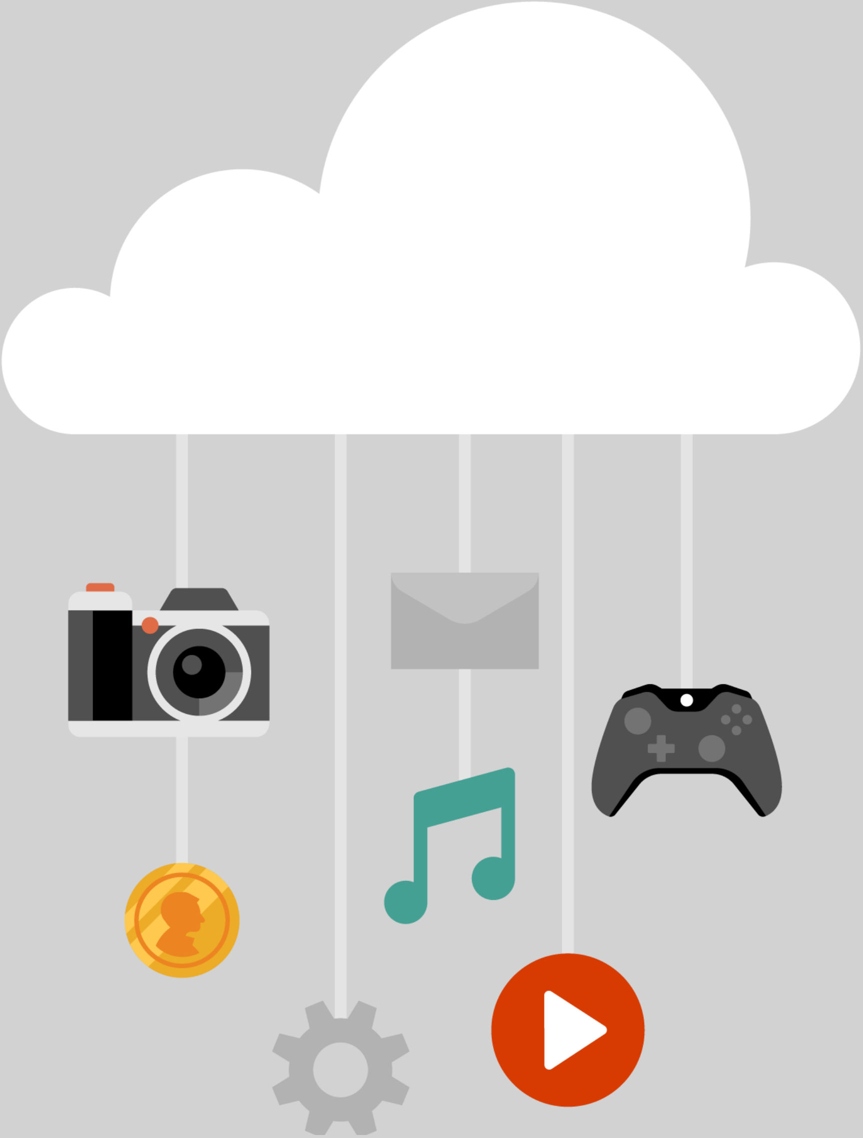
Microsoft

Cloud Computing

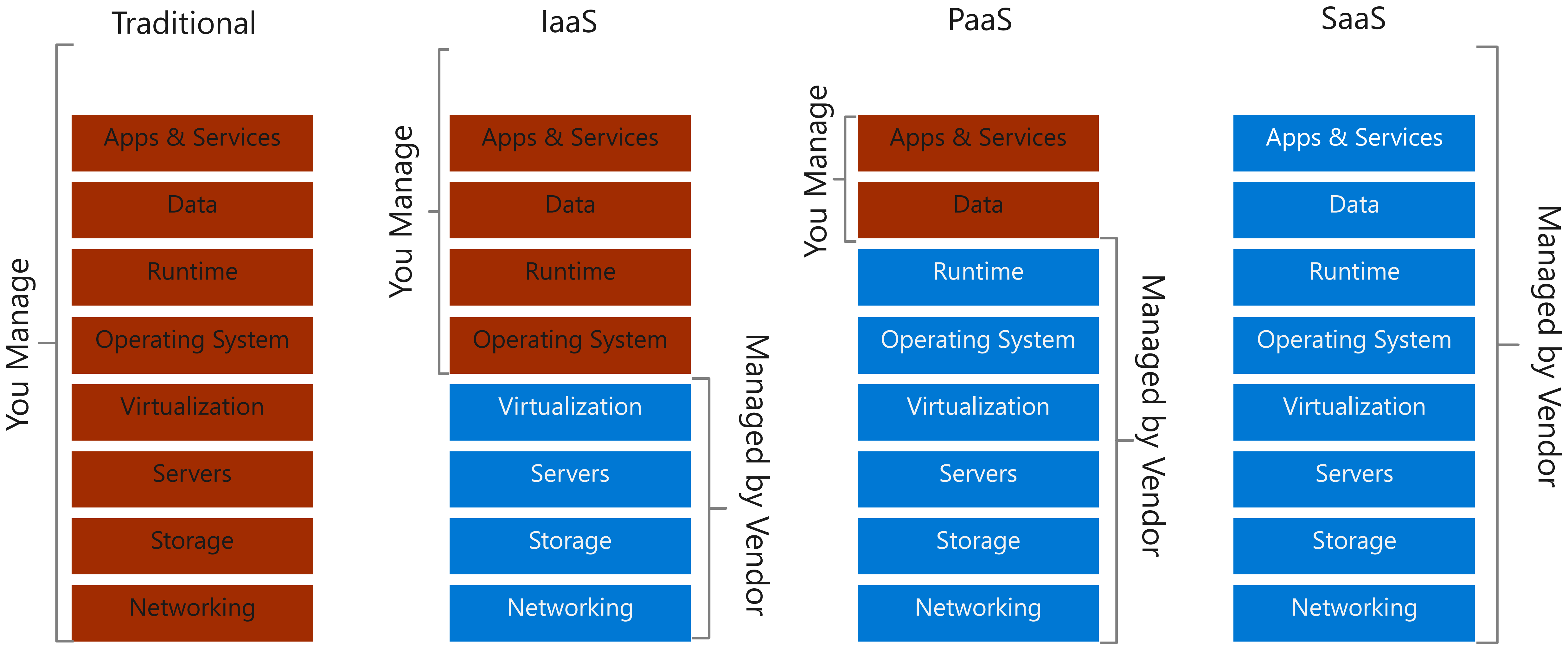
...and how Python fits in ;)



Objectives

1. Review Standards and industry acceptance
2. Understand Cloud Computing terms and concepts
3. Differentiate workloads that are ideal for Cloud Computing

Cloud Models



Traditional	IaaS	PaaS	SaaS
Apps & Services	Apps & Services	Apps & Services	Apps & Services
Data	Data	Data	Data
Runtime	Runtime	Runtime	Runtime
Operating System	Operating System	Operating System	Operating System
Virtualization	Virtualization	Virtualization	Virtualization
Servers	Servers	Servers	Servers
Storage	Storage	Storage	Storage
Networking	Networking	Networking	Networking

Cloud Virtual Machines
On-premises virtual machines

Cloud Services

Office 365, Gsuite, ...

Evolving Hosting Options

Legacy

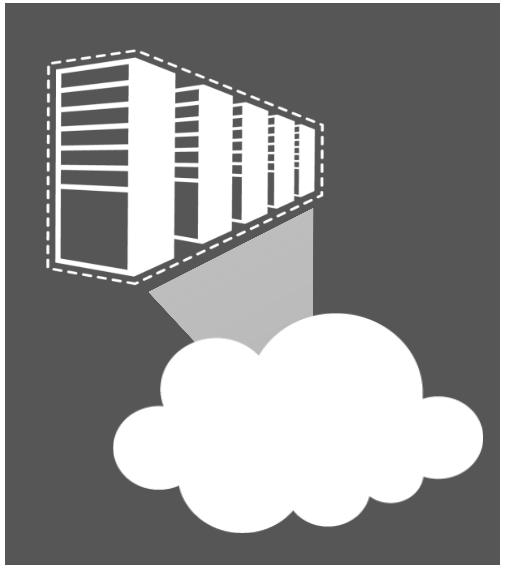
Physical

- 47 percent of new apps are on-premises
- 88 percent of sockets in corporate data center

Past

Virtual

- 98 percent of large organizations have some degree of virtualization



Present

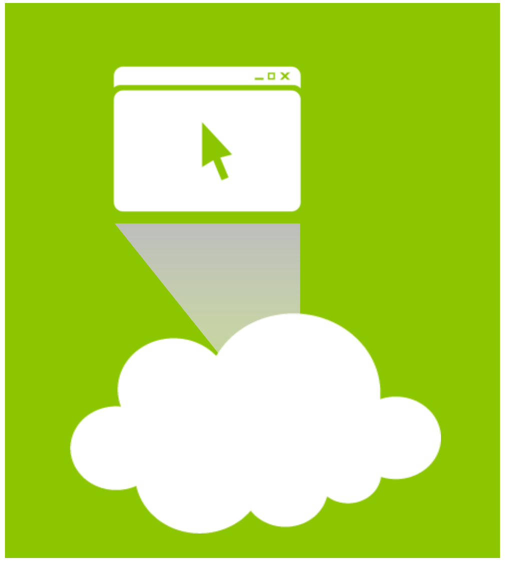
IaaS

- 20 percent of organizations have private clouds
- Majority of cloud growth is IaaS

Present

PaaS

- Majority of new cloud apps are Product as a Service (PaaS)
- Most efficient model for cloud development



SaaS

- About 16 percent of new apps qualify as Software as a Service (SaaS)
- Business model, not hosting model; there are on-premises SaaS apps

Service Models – Infrastructure as a Service (IaaS)

Consumer can create new virtual machines

Consumer manages their virtual machines

Virtualization layer managed by provider

Service Models – Platform as a Service (PaaS)

Platform (e.g. Database servers, Kubernetes clusters, …) managed by provider

Customer manages resources on the platform

- Databases
- Docker containers
- Web Applications

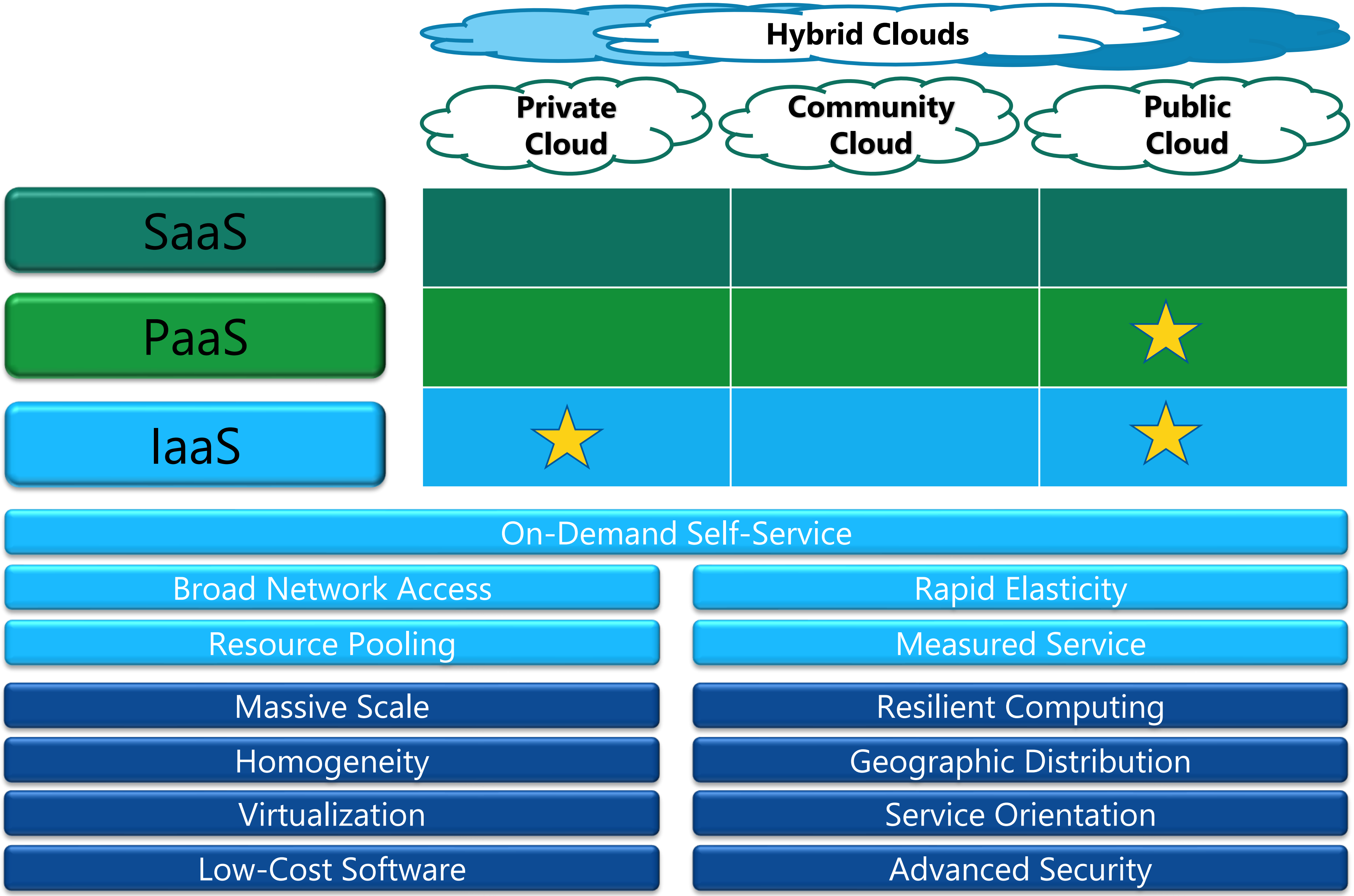
Service Models – Software as a Service (SaaS)

Vendor manages entire product

Customer only accesses the final product

Very commonplace nowadays

- GitHub, Azure DevOps, BitBucket
- Binder
- Teams (usually part of Microsoft 365)
- G Suite (Google’s alternative to Microsoft 365)
- …



Deployment Models

Hybrid Clouds

Service Models

	Private Cloud	Community Cloud	Public Cloud
SaaS			
PaaS			★
IaaS	★		★

Essential Characteristics

On-Demand Self-Service

Broad Network Access	Rapid Elasticity
Resource Pooling	Measured Service

Common Characteristics

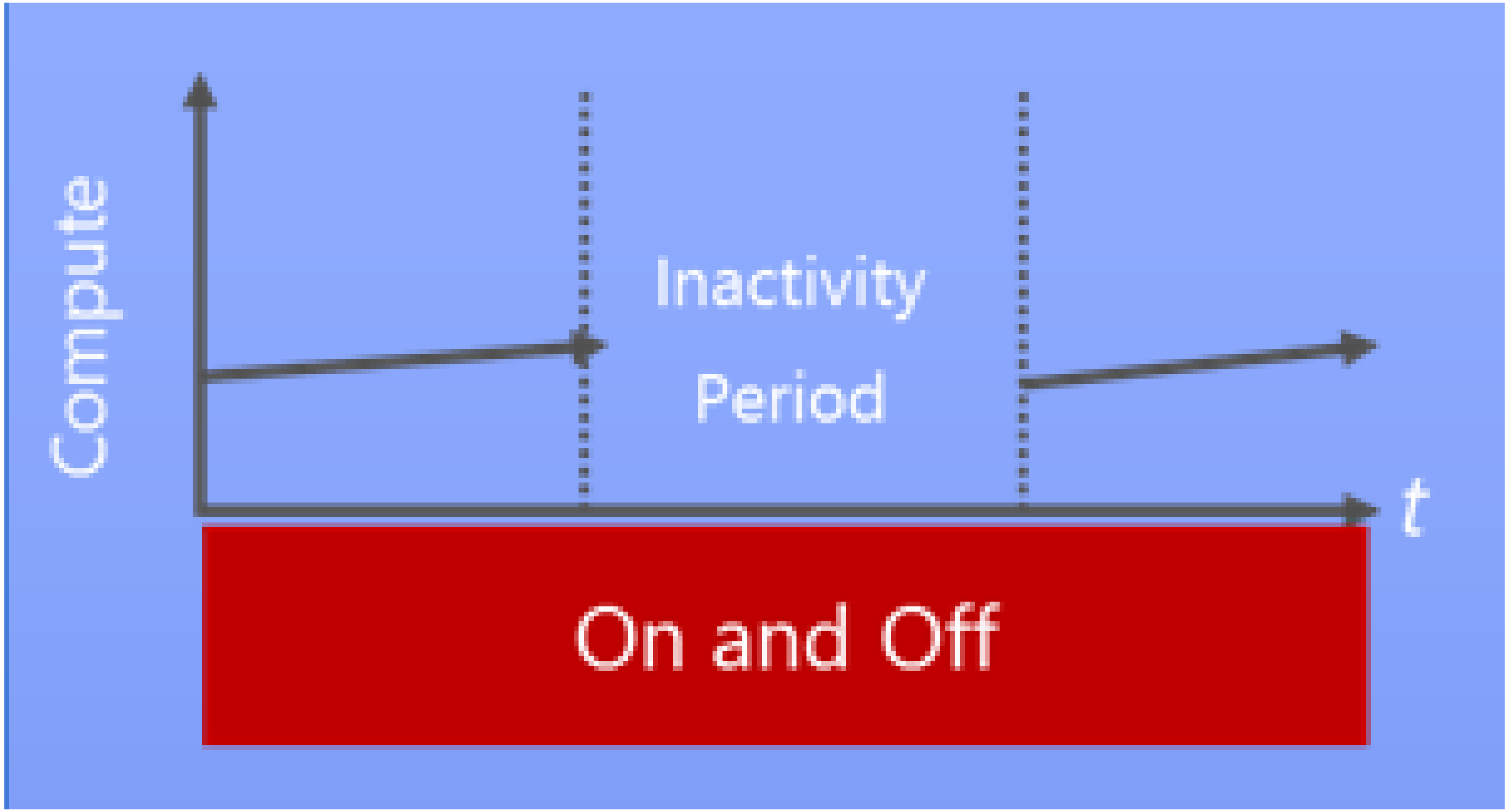
Massive Scale	Resilient Computing
Homogeneity	Geographic Distribution
Virtualization	Service Orientation
Low-Cost Software	Advanced Security

What workloads are best suited for migration?

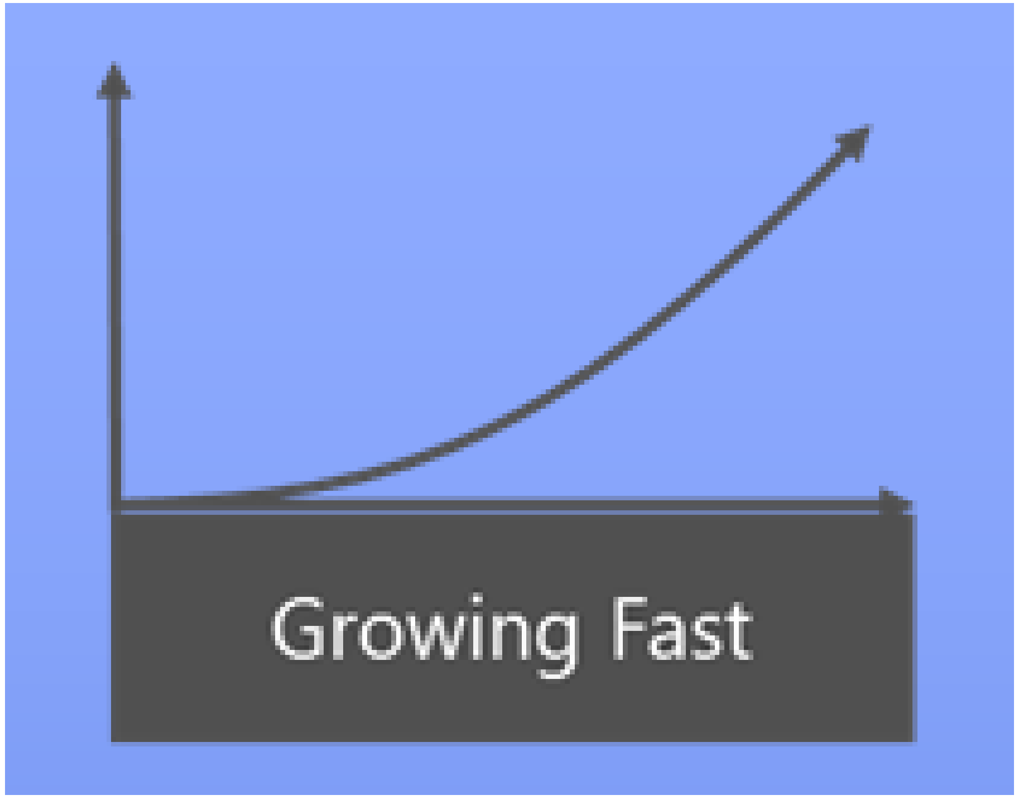
Based on :

- Demand Patterns
- Application Architecture

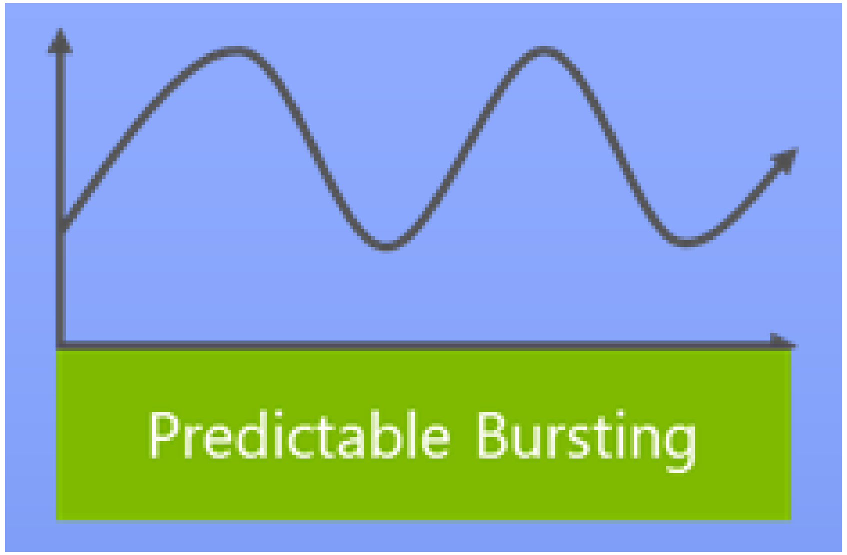
On-and-Off

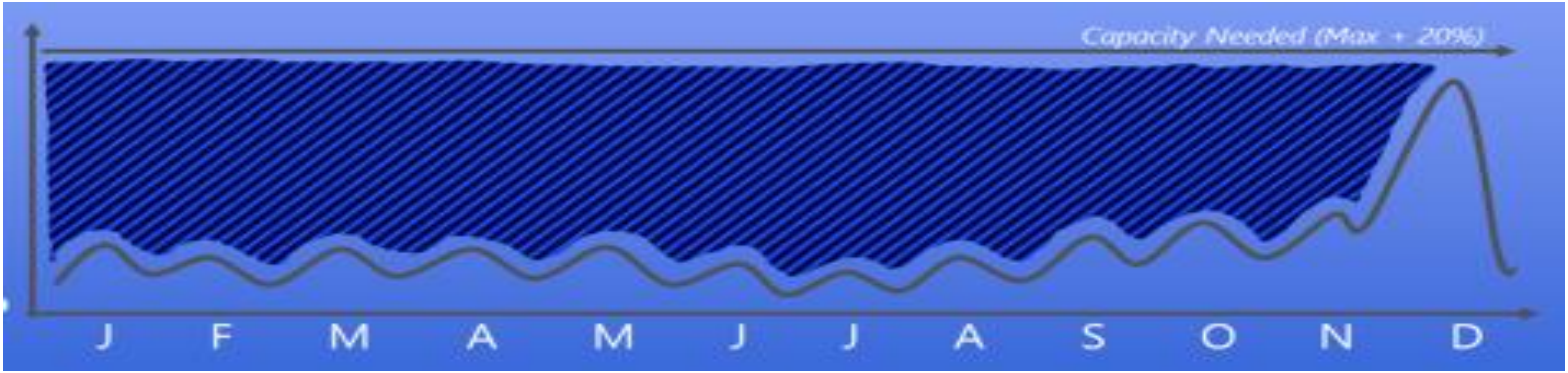


Growing Fast

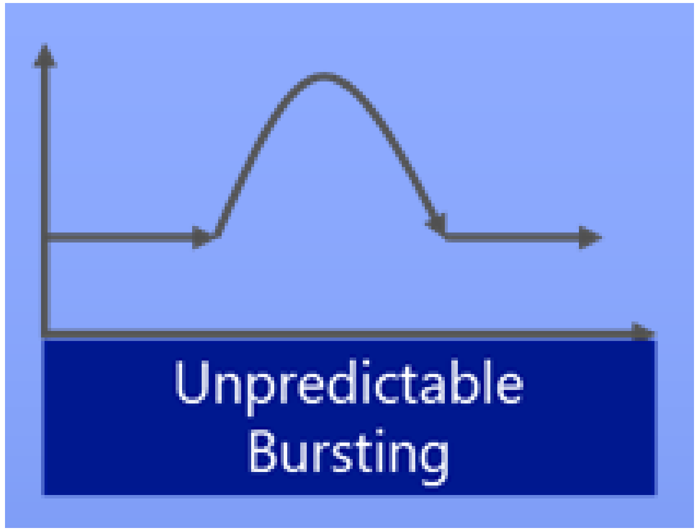


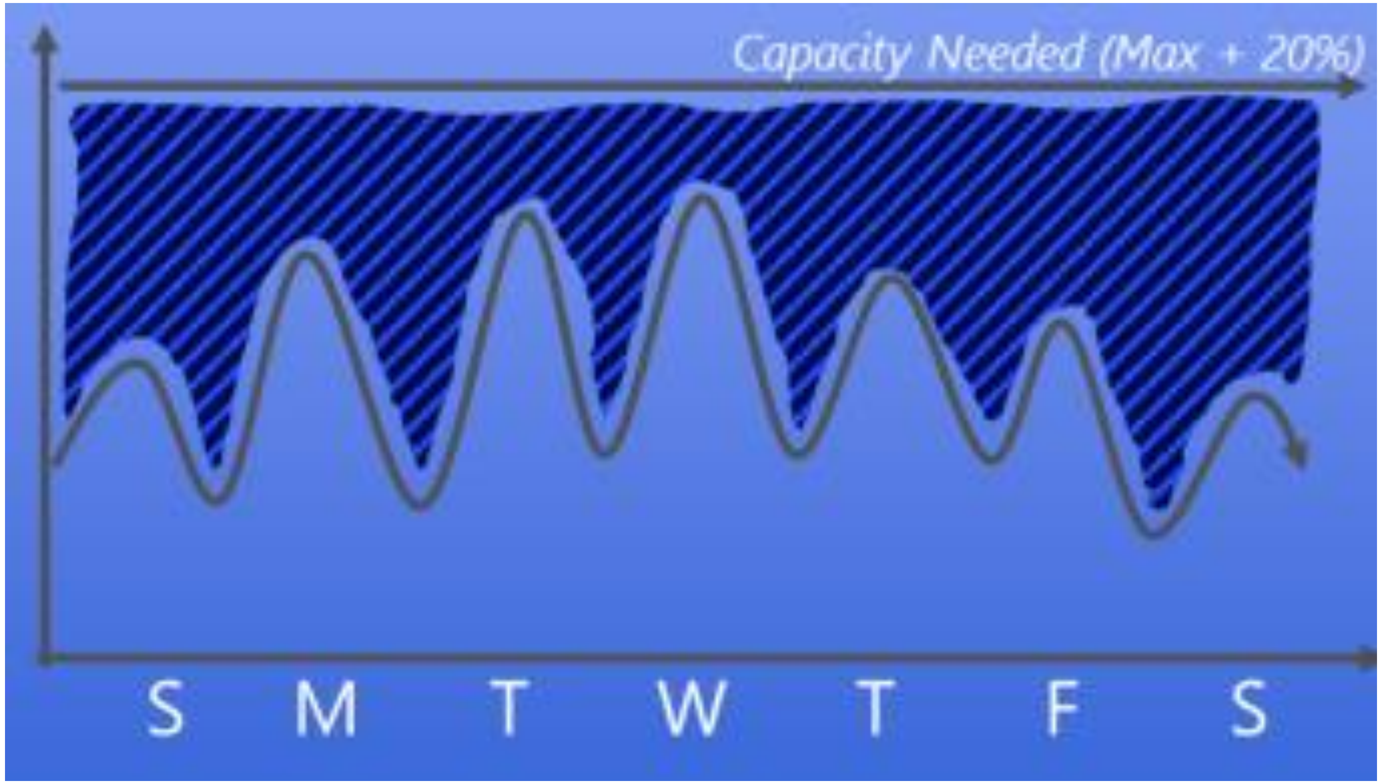
Predictable Bursting



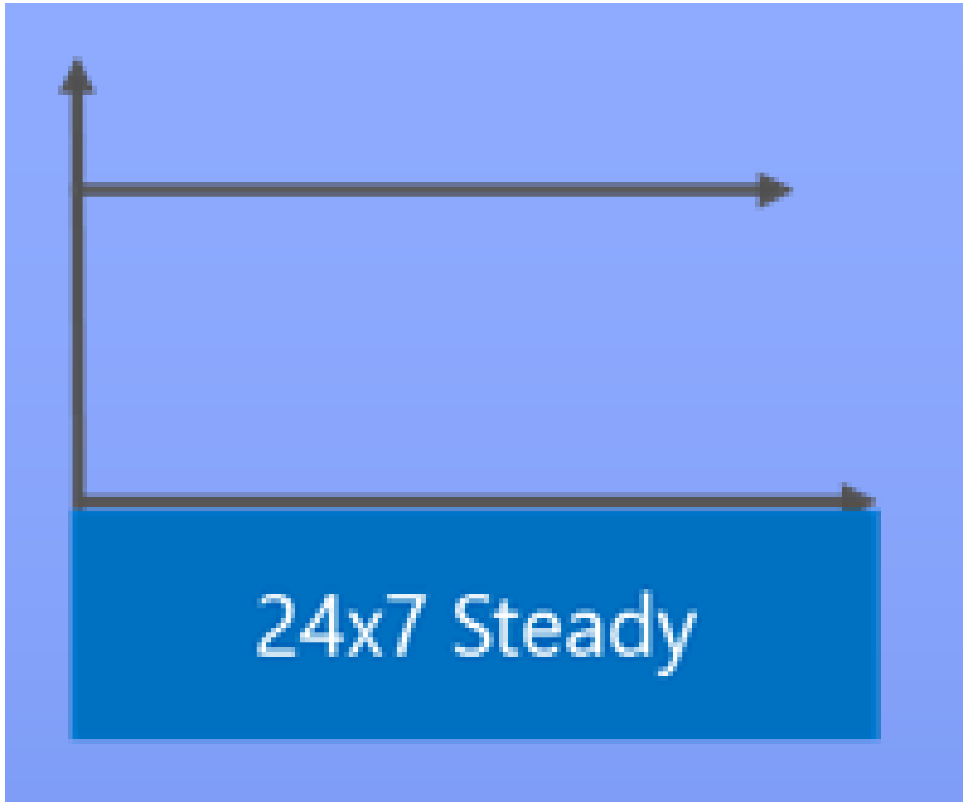


Unpredictable Bursting

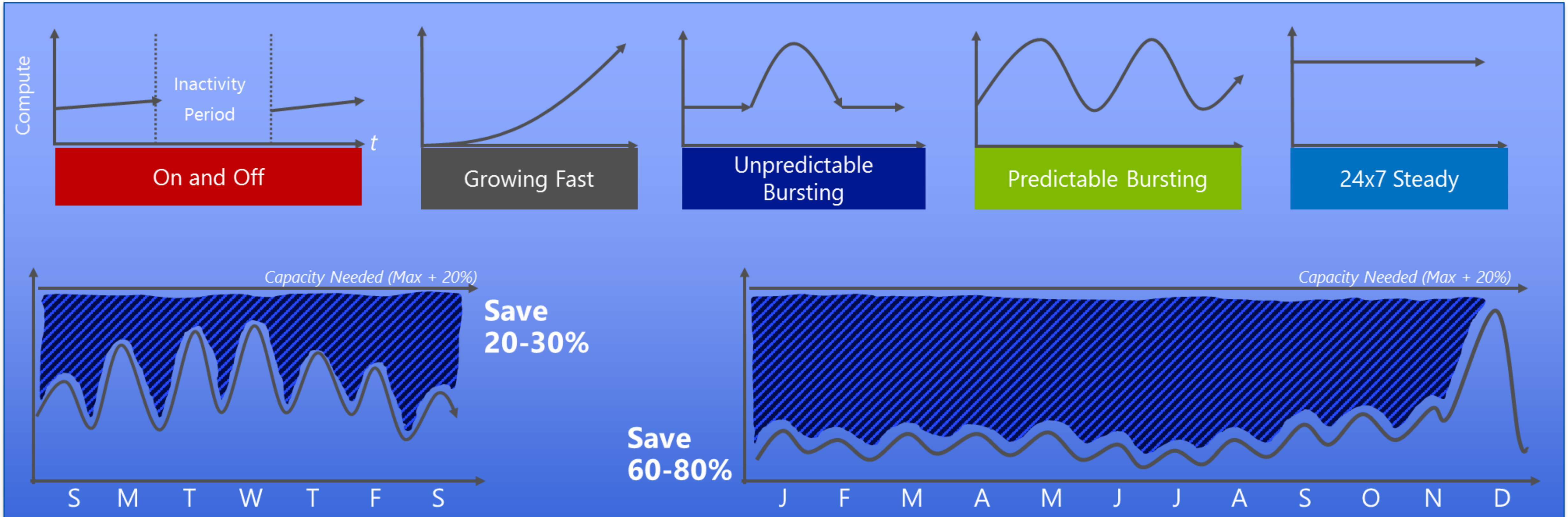




Always Steady



Computing Demand Patterns



On and Off (30%) | Growing Fast (15%) | Unpredictable Bursting (25%) | Predictable Bursting (20%) | 24x7 Steady (10%)

Your Application Portfolio – What Does it Look Like..?

How does Python fit in?

- Python can utilize cloud resources
  - Store data in a hosted database
  - Make use of compute-heavy Machine Learning and Artificial Intelligence solutions
  - Submit jobs for batch processing (“churn the numbers”)
- Python apps can be hosted on PaaS services
- Access to AWS, Azure and GCP via SDK
  - **S**oftware **D**evelopment **K**it
  - Provides classes and functions to interact with a service (here: a cloud provider)
- SDK wraps around standard REST API calls
  - Programming **can** be done without SDK, but is unnecessarily complex