CLOUD COMPUTING APPLICATIONS

CLOUD COMPUTING INTRODUCTION

Roy Campbell & Reza Farivar

Tremendous Buzz

"Not only is it faster and more flexible, it is cheaper. […] the emergence of cloud models radically alters the cost-benefit decision"

(FT)

"Cloud computing achieves a quicker return on investment"

(Lindsay Armstrong of salesforce.com)

"In an economic downturn, the appeal of that cost advantage will be greatly magnified"

(IDC)

"Revolution, the biggest upheaval since the invention of the PC in the 1970s […] IT departments will have little left to do once the bulk of business computing shifts […] into the cloud"

(Nicholas Carr)

"No less influential than e-business"

(Gartner)

"The economics are compelling, with business applications made three to five times cheaper and consumer applications five to 10 times cheaper"

(Merrill Lynch)

"Domestic cloud computing estimated to grow at 53%"

(moneycontrol.com)

Perils of Corporate Computing

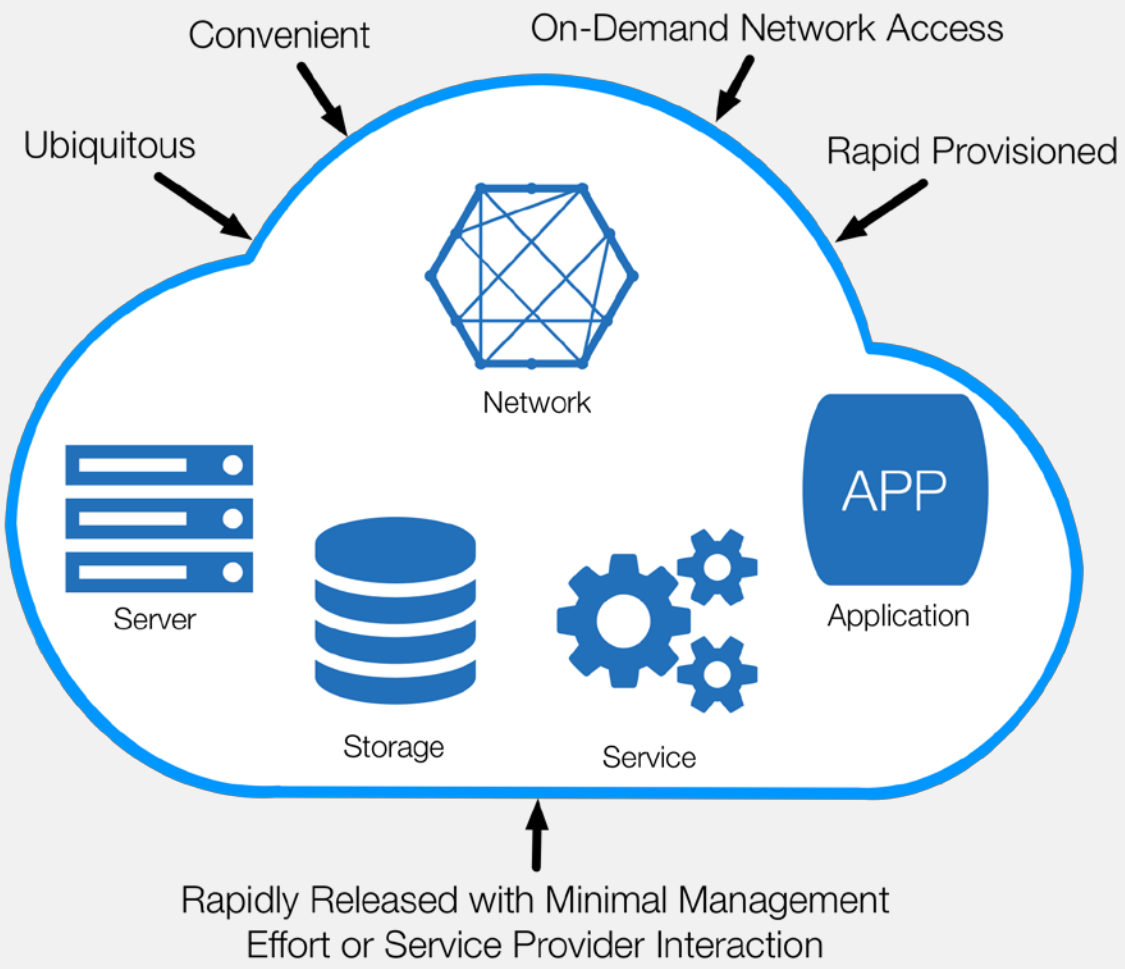
- Own information systems ☺
- However
  - Capital investment ☹
  - Heavy fixed costs ☹
  - Redundant expenditures ☹
  - High energy cost, low CPU utilization ☹
  - Dealing with unreliable hardware ☹
  - High levels of overcapacity (technology and labor) ☹
- NOT SUSTAINABLE

Back to the Future

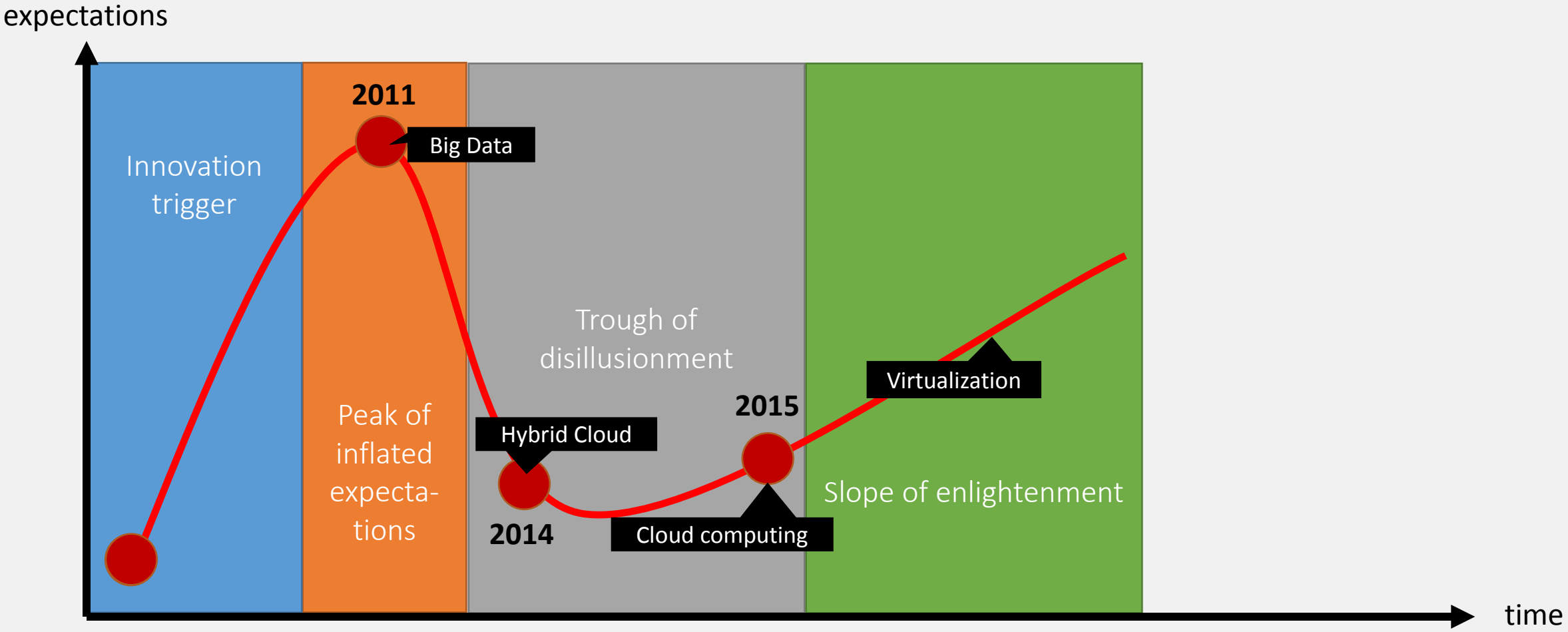
“Computing may someday be organized as a public utility, just as the telephone system is organized as a public utility”

(John McCarthy, 1961)

Cloud Computing



Cloud Adoption: Gartner's Hype Cycle



Delivery Models

- Software as a Service (SaaS)
  - Use provider's applications over a network
  - SalesForce.com
- Platform as a Service (PaaS)
  - Deploy customer-created applications to a cloud
  - AppEng
- Infrastructure as a Service (IaaS)
  - Rent processing, storage, network
  - Capacity and other fundamental computing resources
  - EC2, S3

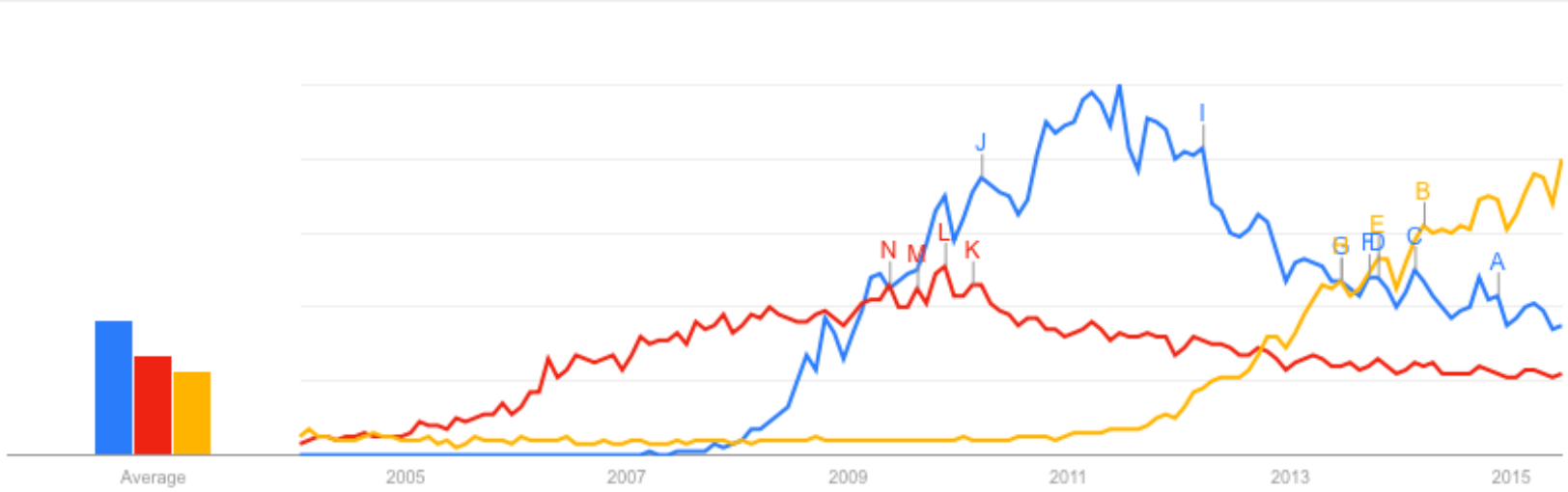
Synergy: Cloud Computing, Virtualization, and Big Data

Cloud Computing
Search term

Virtualization
Search term

Big Data
Search term

Interest over time



Google trends

Big Data Data Revolution and Clouds

- Data collection too large to transmit economically over Internet -- Petabyte data collections

- Computation is data intensive
  - Lots of disks, networks and CPUs
  - Overhead of maintaining cyber infrastructure is expensive
  - Users buy Big Data services from Clouds to share overhead

- Easy-to-write programs, fast turnaround
- MapReduce – Hadoop, PIG, HDFS, HBase

CLOUD COMPUTING APPLICATIONS

CLOUDONOMICS: PART 1

Roy Campbell & Reza Farivar

Cloudonomics: Part 1

Economics necessitates Cloud computing:

- Part 1: Utility Pricing
- Part 2: Benefits Common Infrastructure

See other details and benefits in

“Cloudonomics: A Rigorous Approach to Cloud Benefit Quantification,” Joe Weinman

https://www.csiac.org/sites/default/files/journal_files/stn14_4.pdf

Value of Utility Pricing

- Cloud services don’t need to be cheaper to be economic!
- Consider a car
  - Buy or lease for $10 per day
  - Rent a car for $45 a day
  - If you need a car for 2 days in a trip, buying would be much more costly than renting
  - **It depends on the demand**

Utility Pricing in Detail

D(t): demand for resources, 0<t<T

P = max(D(t)): Peak Demand; A = Avg (D(t)): Average Demand

B = Baseline (owned) unit cost; B_T = Total Baseline Cost

C = Cloud unit cost; C_T = Total Cloud Cost

U = C / B : Utility Premium (for the rental car example, U = 4.5)

C_T

(because the Baseline should handle Peak Demand)

When is the Cloud cheaper than owning?

Substituting for C_T, B_T :

which implies

i.e., when Utility Premium is less than ratio of Peak Demand to Average Demand

Utility Pricing in Real World

- In practice, demands are often highly spiky
  - News stories, marketing promotions, product launches, Internet flash floods (Slashdot effect), tax season, Christmas shopping, etc.
- Often a hybrid model is the best
  - You own a car for daily commute, and rent a car when traveling or when you need a van to move
  - Key factor is again the ratio of Peak Demand to Average Demand
  - But we should also consider other costs
    - Network cost (both fixed costs and usage costs)
    - Interoperability overhead
    - Consider reliability, accessibility

Summary

- Utility Pricing is good when demand varies over time, as is the case of a start-up or a seasonal business
- When Utility Premium is less than ratio of Peak Demand to Average Demand, Cloud computing is beneficial
- Next, we look at the possible savings that Cloud providers can create using statistical multiplexing

CLOUD COMPUTING APPLICATIONS

CLOUDONOMICS: PART 2

Roy Campbell & Reza Farivar

Cloudonomics: Part 2

Economics necessitates Cloud Computing:

- Part 1: Utility Pricing
- Part 2: Benefits Common Infrastructure

See other details and benefits in

“Cloudonomics: A Rigorous Approach to Cloud Benefit Quantification,” Joe Weinman

https://www.csiac.org/sites/default/files/journal_files/stn14_4.pdf

The Value of Common Infrastructure

- For infrastructure built to peak requirements: Multiplexing demand → higher utilization
  - Lower cost per delivered resource than unconsolidated workloads
- For infrastructure built to less than peak: Multiplexing demand → reduce the unserved demand
  - Lower loss of revenue or a Service-Level agreement violation payout

A Useful Measure of “Smoothness”

The coefficient of variation:

$$C_v = \frac{\text{standard deviation } \sigma}{\text{mean } |\mu|}$$

C_v is a measure of smoothness

- small is smooth!
- large mean and/or smaller standard deviation

Implications of “Smoothness”

- A fixed-asset facility servicing highly variable jobs yields low utilization
- Same facility servicing smooth jobs yields high utilization
- **Multiplexing jobs with different distributions may reduce the coefficient of variation C_V**

Case Study of C_V for Independent Jobs

- X_1, X_n,…, X_n independent jobs with standard variation σ and mean μ
- Aggregated jobs
  - Mean → sum of means: $n.\ \mu$
  - Variance → sum of variances: $n.\sigma^2$
  - Aggregate C_v→ $\frac{\sqrt{n}.\sigma}{n.\mu} = \frac{\sigma}{\sqrt{n}.\mu} = \frac{1}{\sqrt{n}} C_v$

Case Study of C_V for Independent Jobs

Adding *n* independent jobs reduces C_v by $1/\sqrt{n}$

- Penalty of insufficient/excess resources grows smaller
- Aggregating 100 workloads brings the penalty to 10%

Case Study of C_V for Correlated Jobs

- Best Case: Negative correlation
  - Optimal packing of customer jobs
  - X and 1-X → Sum is 1, $C_v = 0$
  - Optimally smooth, best CPU utilization

- Worst Case: Positive correlation
  - Mean: n.μ(X), standard deviation: n.σ(X)
  - Aggregate $C_v = C_v(X) = \frac{\sigma(X)}{\mu(X)}$
  - Which isn't smoother!

Results from Theory

- Negative-correlated jobs
  - Private, mid-size, and large-size providers can experience similar statistics of scale
- Independent jobs
  - Mid-size providers can achieve similar statistical economies to an infinitely large provider

Common Infrastructure in Real World

- Available data on economy of scale for large providers is mixed
  - Use the same COTS computers and components
  - Locate near cheap power supplies → everyone can do that
  - Early entrant automation tools → 3rd parties take care of it
- Takeaway lesson: you don’t need to be as large as Amazon.com to compete! ☺
  - At least according to “Value of Common Infrastructure”

CLOUD COMPUTING APPLICATIONS

BIG DATA

Roy Campbell & Reza Farivar

Big Data (a Singular Phrase)!

- A collection of data sets so large and complex, it’s impossible to process it on one computer with the usual databases and tools

- Because of its size and complexity, Big Data is hard to capture, store, copy, delete (privacy), search, share, analyze, and visualize

Big Data

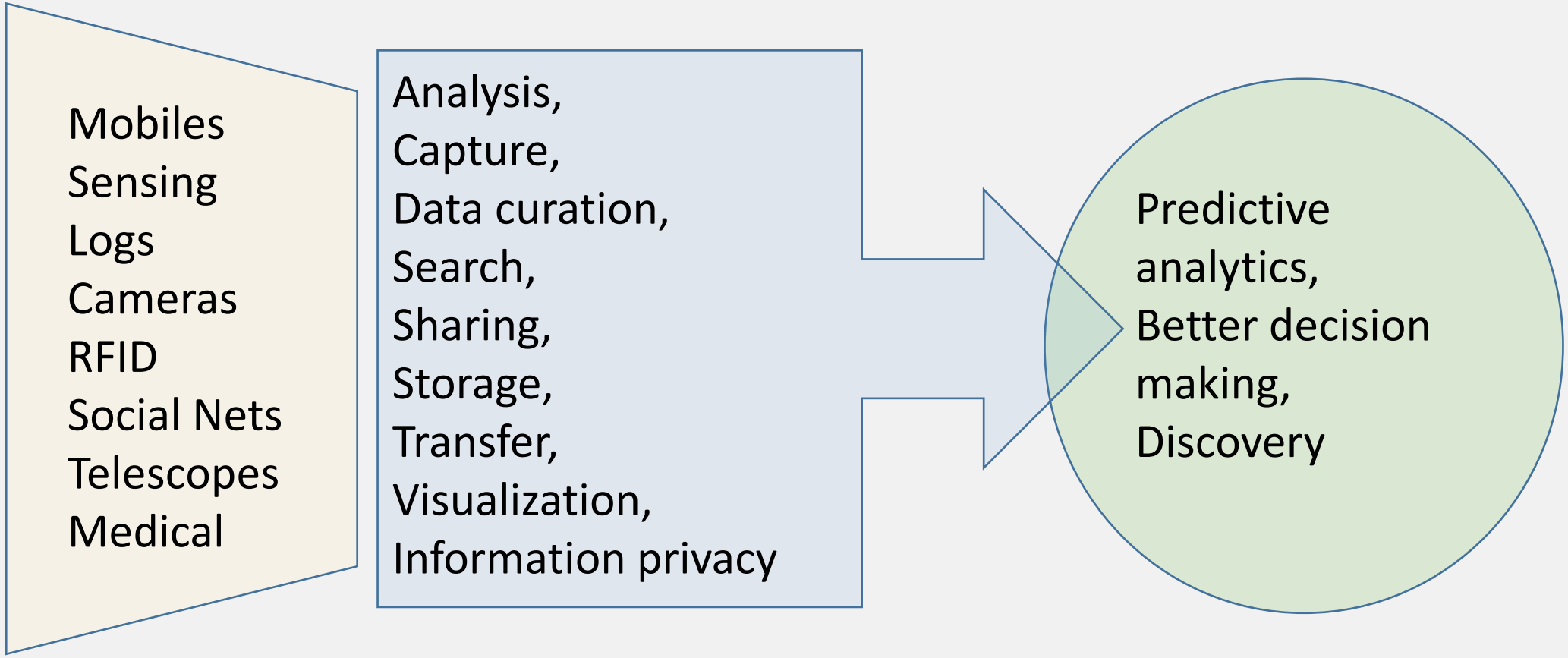
Big Data represents the information assets characterized by such high

- Volume,
- Velocity, and
- Variety

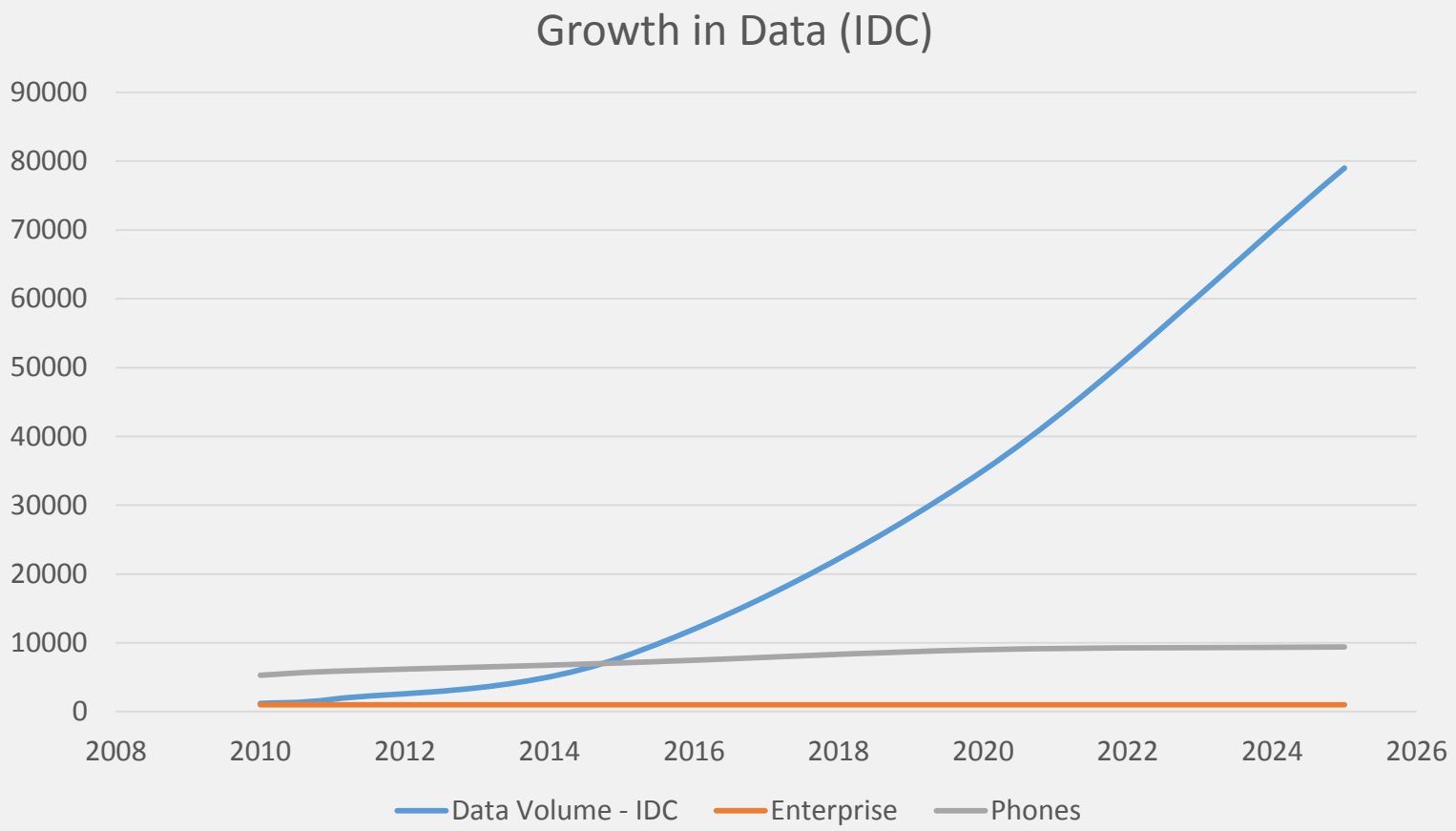
as to require specific technology and analytical methods for its transformation into

- Value

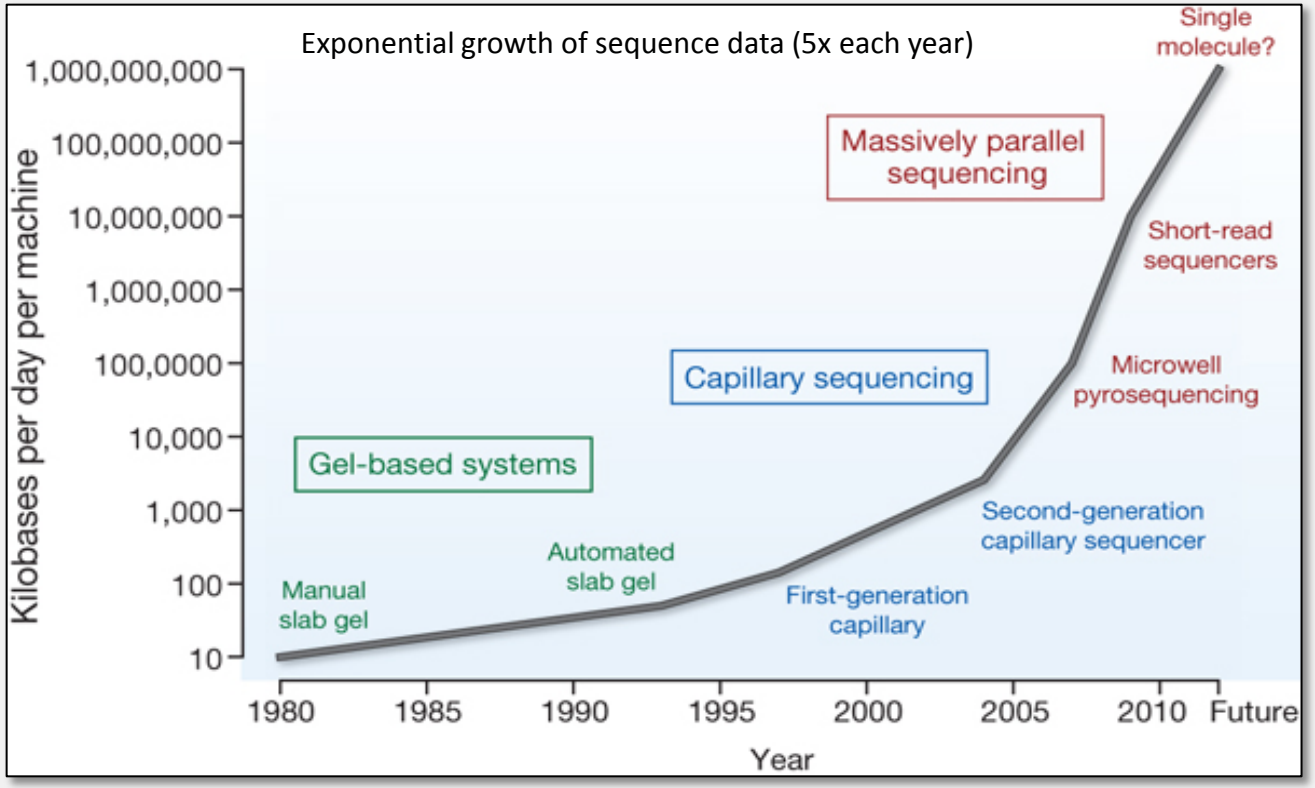
Challenges


Mobiles
Sensing
Logs
Cameras
RFID
Social Nets
Telescopes
Medical
Analysis,
Capture,
Data curation,
Search,
Sharing,
Storage,
Transfer,
Visualization,
Information privacy
Predictive analytics,
Better decision making,
Discovery

Timeline

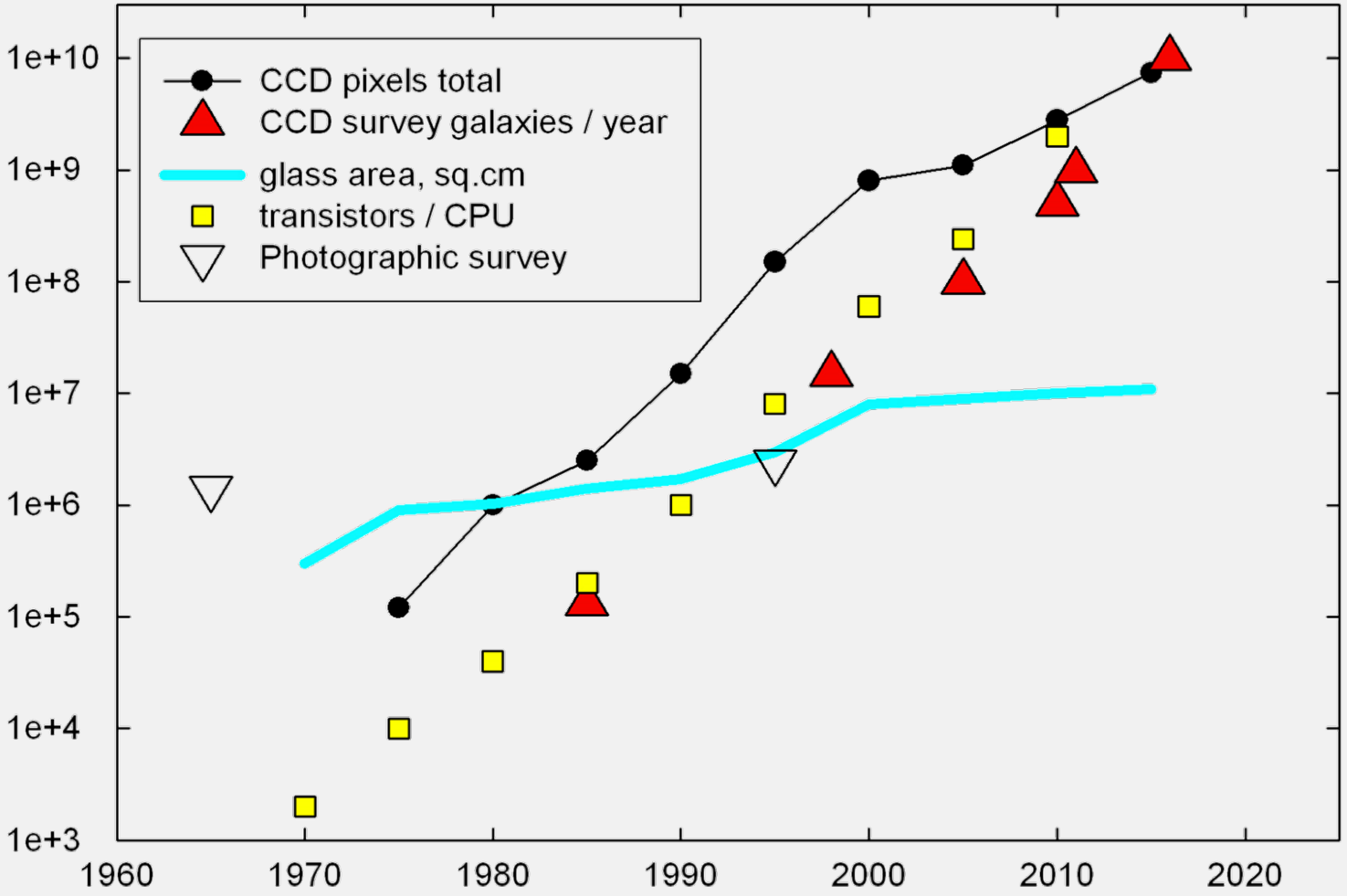

Growth in Data (IDC)
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
2008
2010
2012
2014
2016
2018
2020
2022
2024
2026
Data Volume - IDC
Enterprise
Phones

Example: Bioinformatics

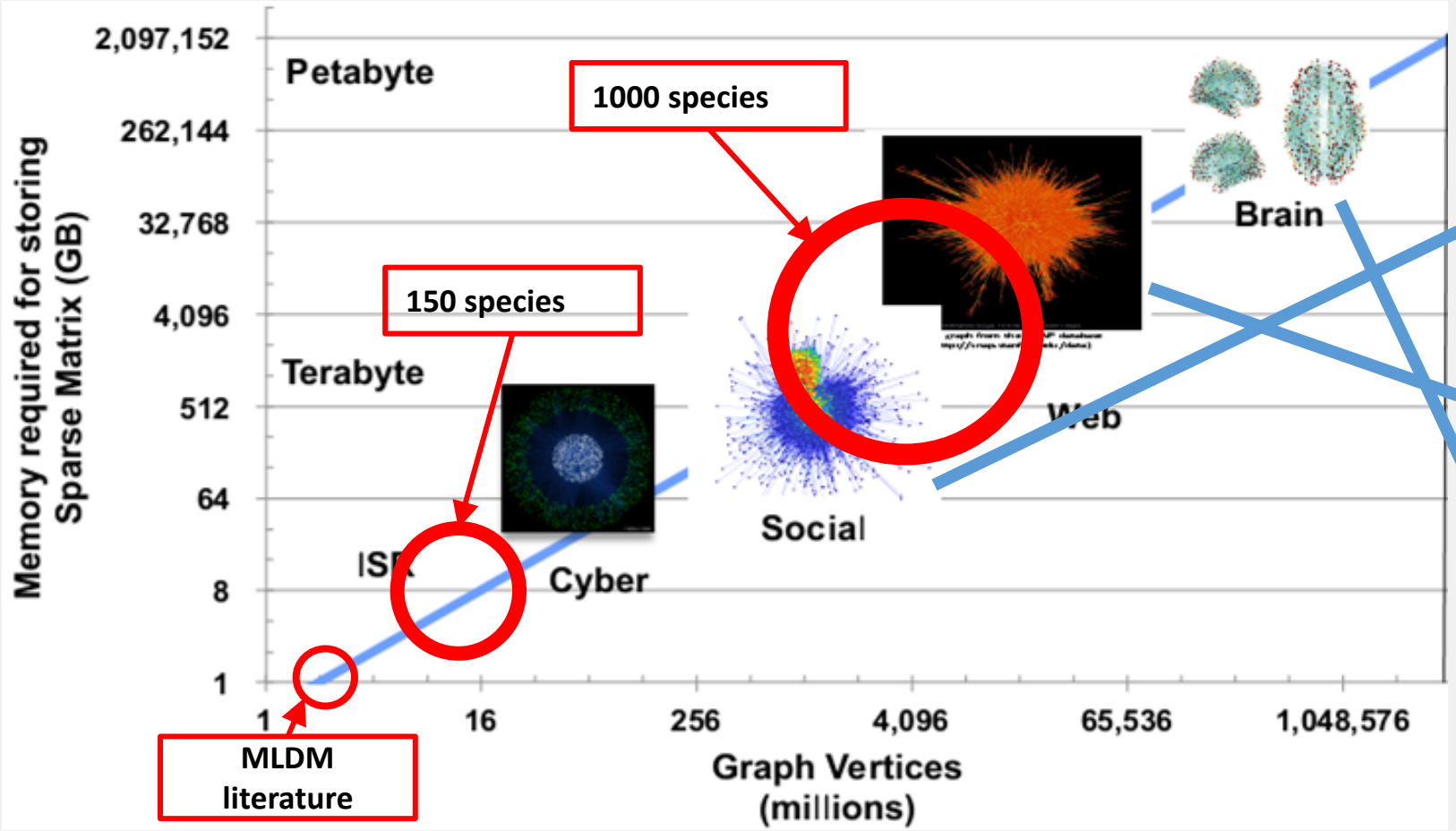


Michael R. Stratton, Peter J. Campbell & P. Andrew Futreal
Nature ***458, 719-724(9 April 2009)***

Example: Astronomy



Example: Graphs



Social scale …
1 billion vertices, 100 billion edges

2.92 TB adjacency list

Web scale …
50 billion vertices, 1 trillion edges

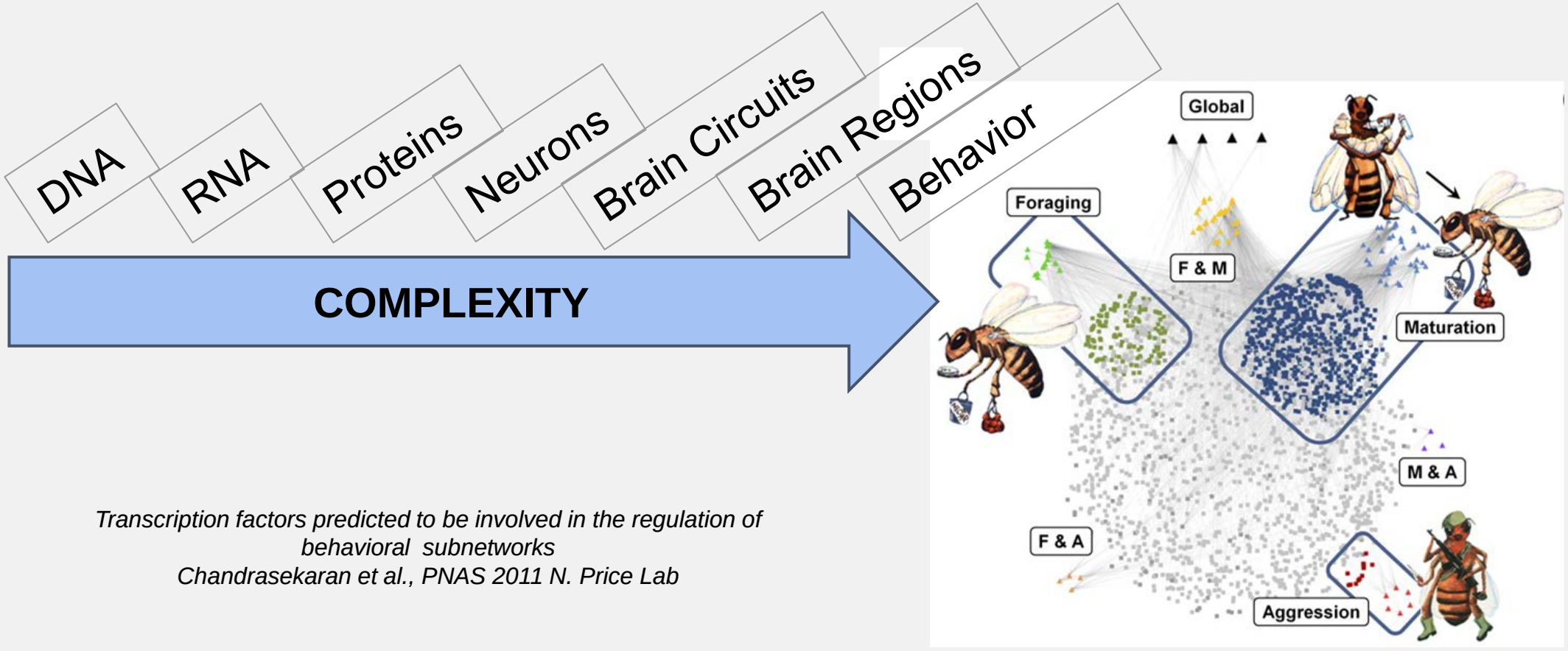
29.5 TB adjacency list

Brain scale …
100 billion vertices, 100 trillion edges

2.84 PB adjacency list

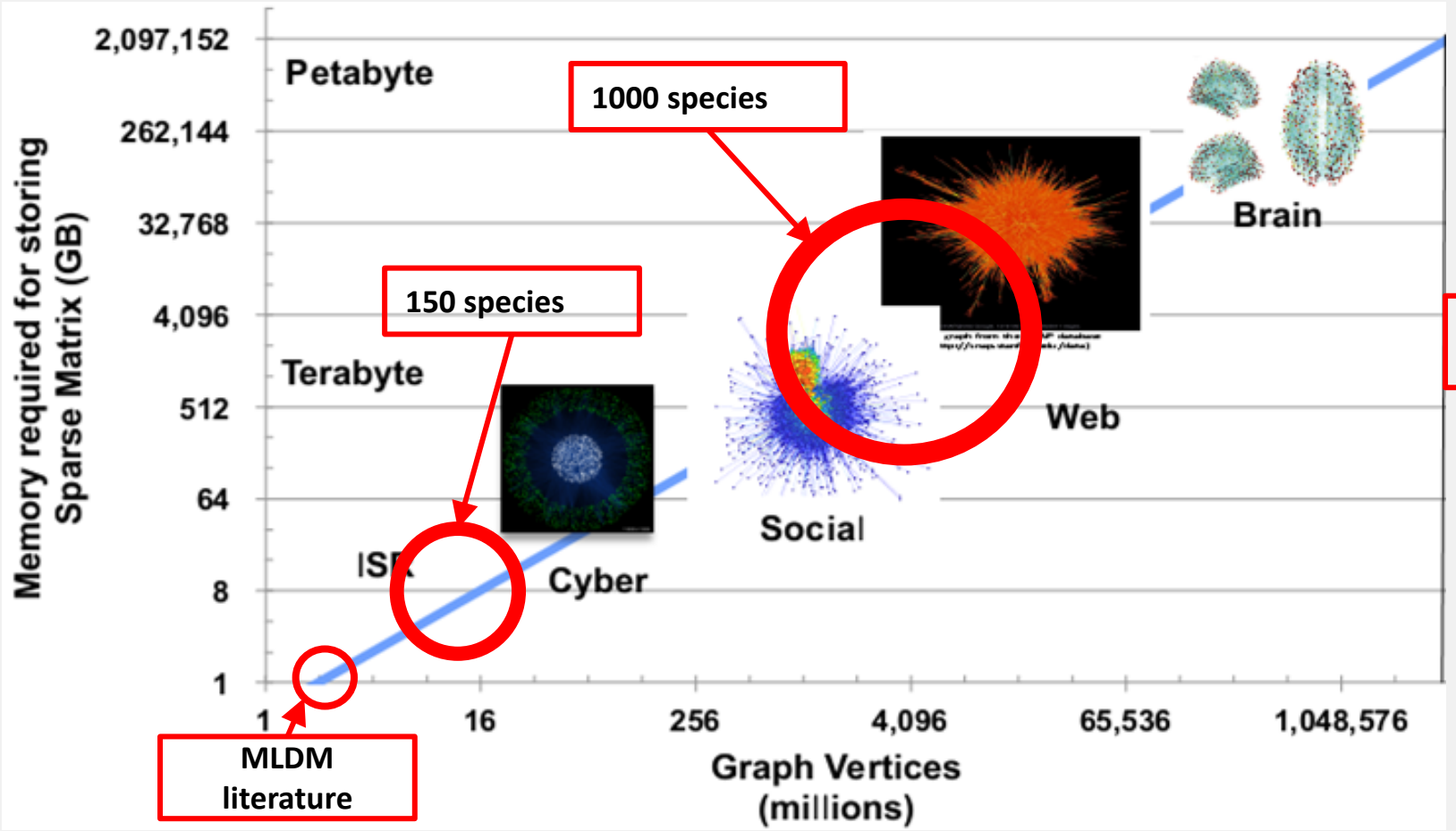
Future: Scale of Real-World Graphs

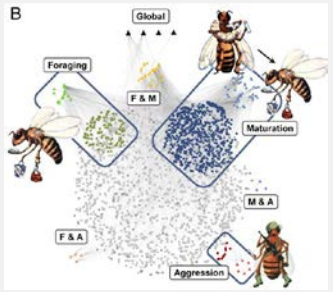
From genes to brains and social behavior…



Transcription factors predicted to be involved in the regulation of behavioral subnetworks
Chandrasekaran et al., PNAS 2011 N. Price Lab

Scale of Real-World Graphs: Today





Genomic and brain to social behavior

CLOUD COMPUTING APPLICATIONS

SUMMARY OF CLOUD INTRODUCTION

Roy Campbell & Reza Farivar

Takeaways

- Multiple reasons for Cloud adoption
- Clouds allow economies
- Sharing for the Big Data revolution?
- Another round in innovation: everyone benefits from Clouds?

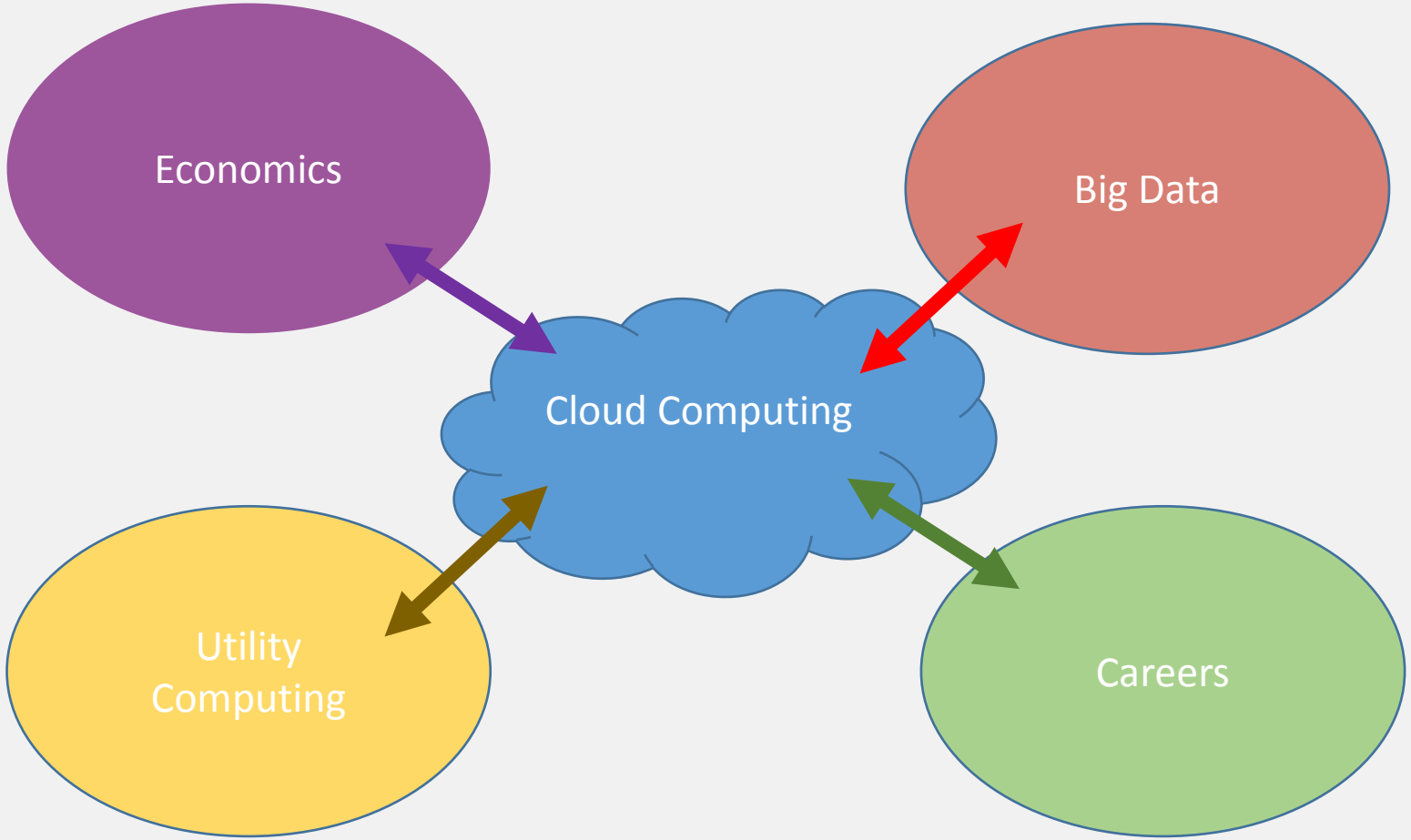
Careers

- Almost all people will use Cloud computing: search, video streaming, social networking
- IT specialists need to know Cloud
  - When to outsource, applications, uses
  - Architectures, sharing, privacy
  - Programming, efficiency, parallelism, scale
- General audience
  - Huge impact on society, business, government
  - New capabilities, policies, ways to communicate
  - Privacy and security

First Exercise

- Show scalability of web service
- Demonstrate user application
- Software-defined architecture
  - Management controls
  - Billing
  - Software modules available
- Distributed computing
  - Reliability, parallelism
- Provides insight to the power of sharing data

Summary


Economics
Big Data
Cloud Computing
Utility Computing
Careers

CLOUD COMPUTING APPLICATIONS

SOFTWARE DEFINED ARCHITECTURE

Roy Campbell & Reza Farivar

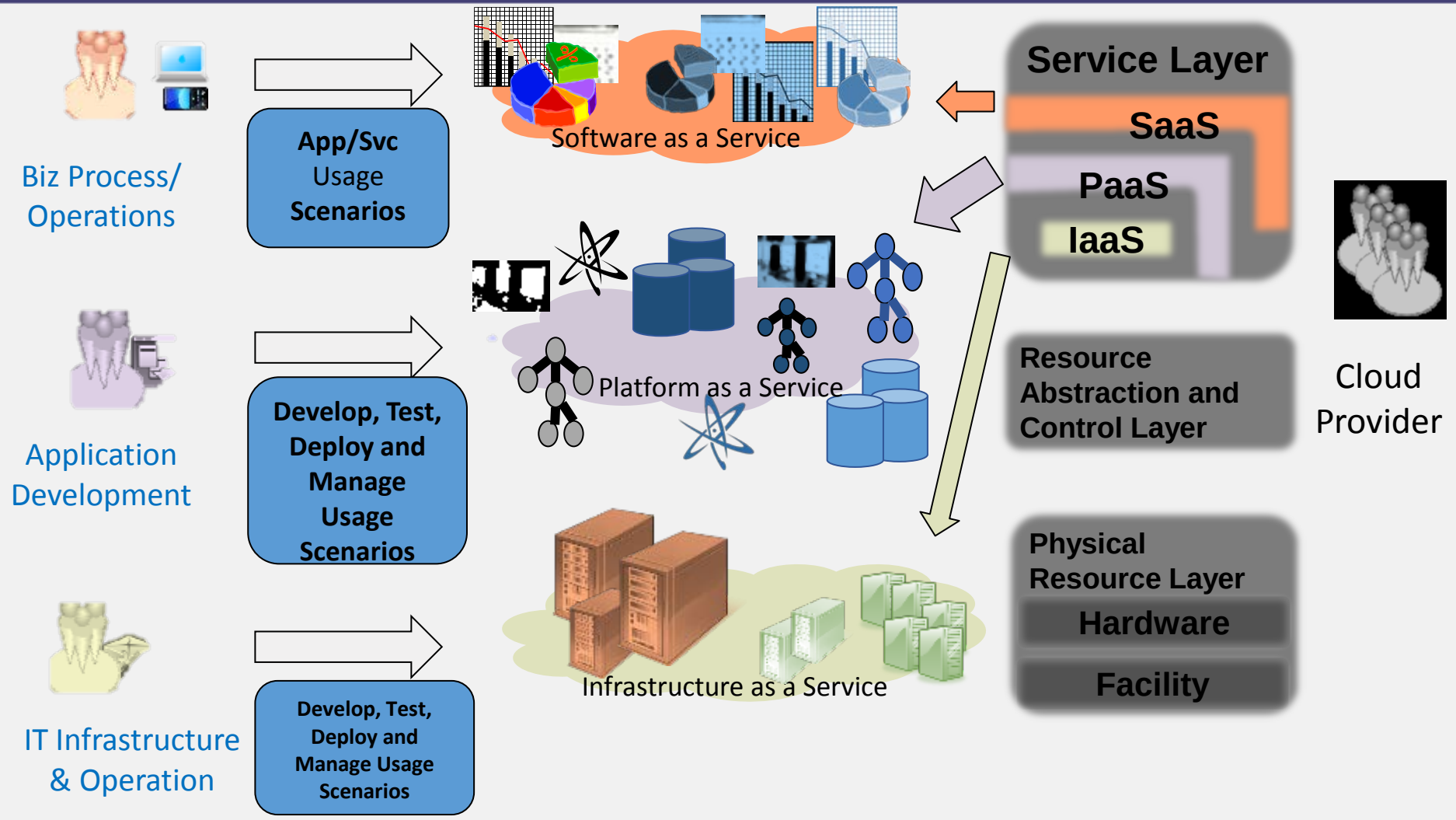
Learning Objectives

- How services are created
- How services can control other services
- The principal architectural components of a cloud and their organization
- How services and orchestration play a role in each layer of a production cloud: IAAS, PAAS, SAAS

Software Defined Architecture

- Cloud provides services, service orchestration, and provisioning
- A Cloud may provide IAAS, PAAS, SAAS and have both internal and external Application Programming Interfaces
- The mechanisms and concept of providing services, orchestration, and provisioning is called a Software Defined Architecture
- A Cloud may contain other software defined entities:
  - Software Defined Network
  - Software Defined Storage
  - Software Defined Compute

Cloud Provider: Service Orchestration

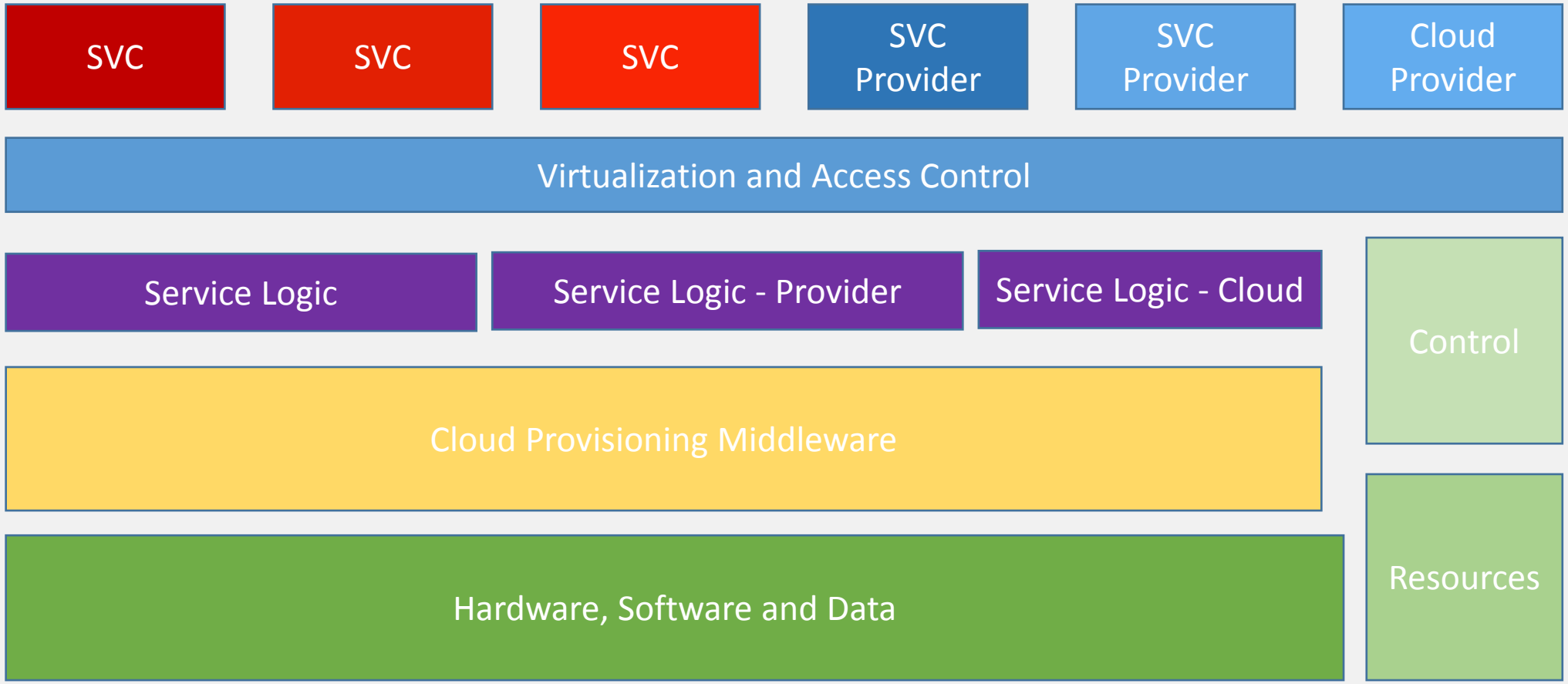


Orchestration

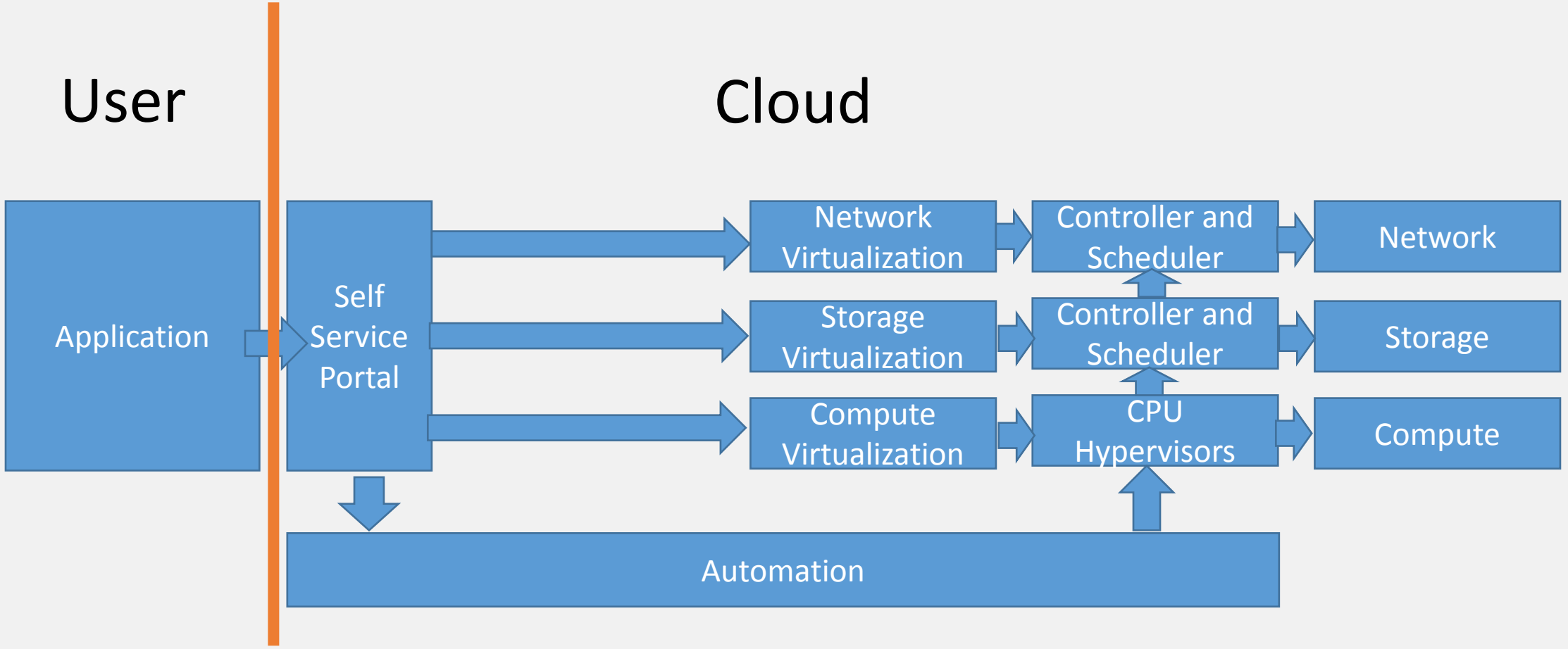
Cloud service orchestration is the:

- *Composing* of architecture, tools and processes used by humans to deliver a defined Service.
- *Stitching* of software and hardware components together to deliver a defined Service.
- *Connecting* and *Automating* of work flows when applicable to deliver a defined Service.
- Provides: up and down scaling, assurance, billing, workflows

Software Defined Architecture



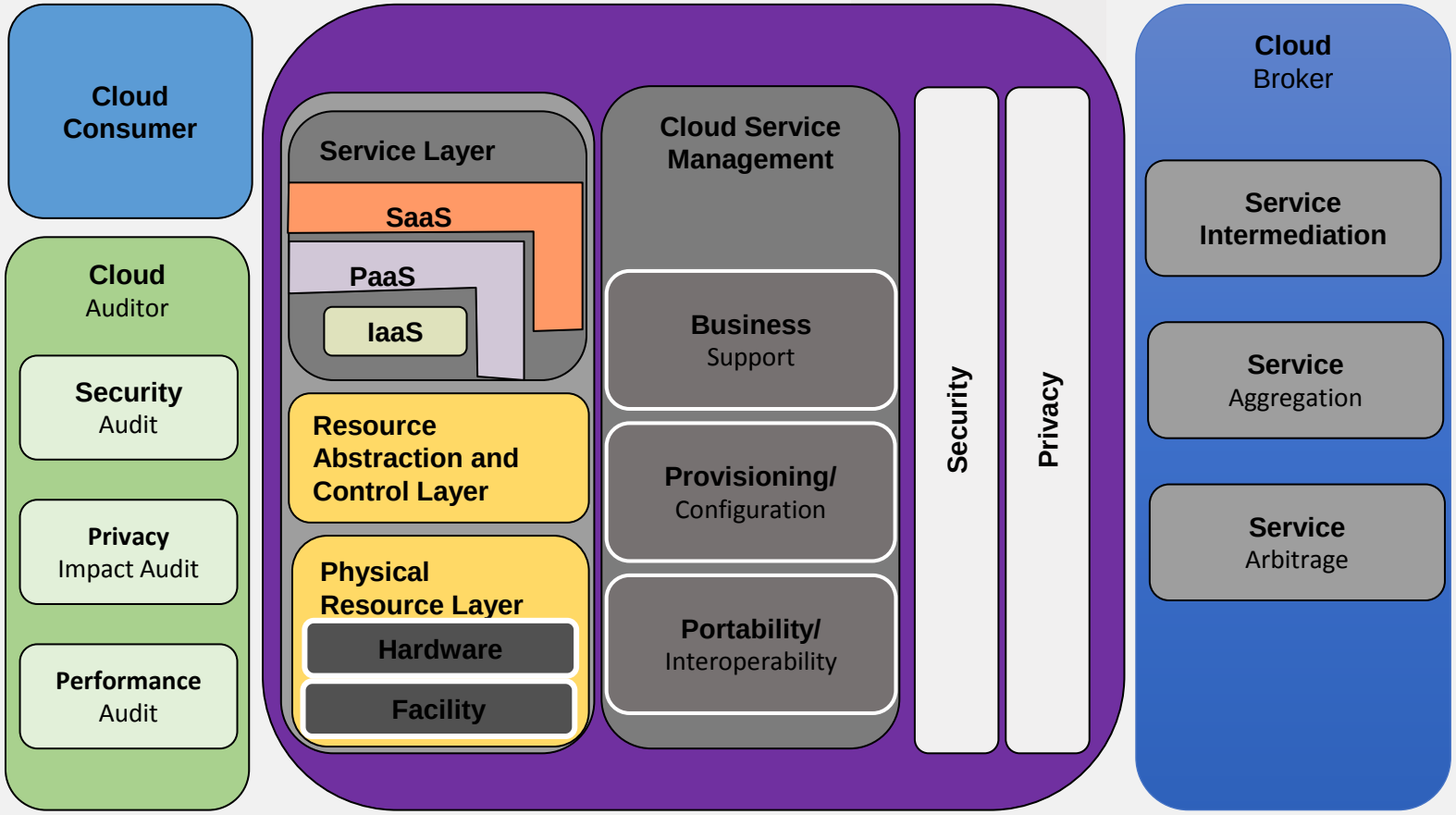
Software Defined Data Center



Content and Learning Objectives

1) Virtualization is a key abstraction in building software defined architectures:
    1) Software Defined Networks
    2) Software Defined Storage
    3) Software Defined Compute
2) Web Service: A Simple Application built on a Data Center
3) Load Balancing: A simple scheme to distribute the load of multiple servers
4) Infrastructure as a Service
5) Mirantis and OpenStack
6) How systems are structured with orchestration

The Combined Conceptual Reference Diagram



CLOUD COMPUTING APPLICATIONS

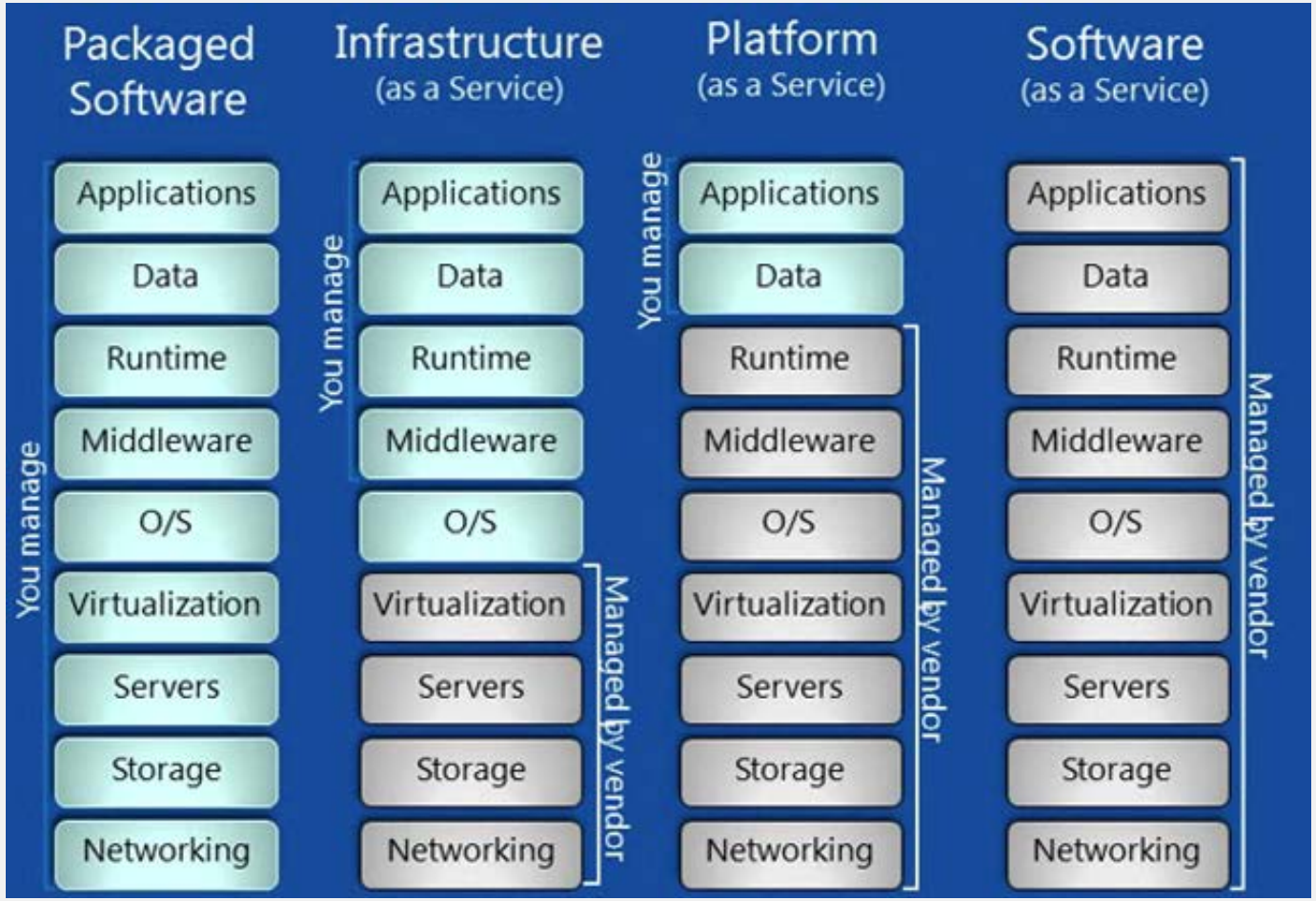
CLOUD SERVICES

Roy Campbell & Reza Farivar

Objective

- Compare IaaS, PaaS, and SaaS
- Look at what services major Cloud companies provide and how they provide them

IaaS, PaaS, and SaaS Comparison



Cloud Fundamentals

- Infrastructure as a Service (IaaS): basic compute and storage resources
  - On-demand servers
  - Amazon EC2, VMWare, vCloud
- Platform as a Service (PaaS): Cloud application infrastructure
  - On-demand application-hosting environment
  - For example, Google AppEngine, Salesforce.com, Windows Azure, Amazon
- Software as a Service (SaaS): Cloud applications
  - On-demand applications
  - For example, GMail, Microsoft Office Web Companions

Platform as a Service (PaaS)

- PaaS is a Cloud computing service that offers a platform for users to run applications on the Cloud

- PaaS is a level above IaaS because unlike IaaS, PaaS does not require users to develop their own operating system environment

Platform as a Service (PaaS)

- Middle ground between SaaS and IaaS
- Development platform
  - Customers use it to develop applications that benefit from the scalability of the Cloud without fully developing their own solution using an IaaS provider
- Offers an application development platform that will automatically scale with demand

The Benefits of the Cloud

The Cloud is about cheap, on-demand capacity

✓ = Managed for You	Standalone Servers	IaaS	PaaS	SaaS
Applications	✗	✗	✗	✓
Runtimes	✗	✗	✓	✓
Database	✗	✗	✓	✓
Operating system	✗	✗	✓	✓
Virtualization	✗	✓	✓	✓
Server	✗	✓	✓	✓
Storage	✗	✓	✓	✓
Networking	✗	✓	✓	✓

Platform as a Service (PaaS)

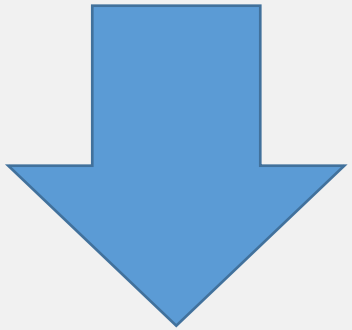
Official definition of PaaS from NIST standard

“The capability provided to the consumer is to deploy, onto the Cloud infrastructure, consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider. The consumer does not manage or control the underlying Cloud infrastructure, including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.”

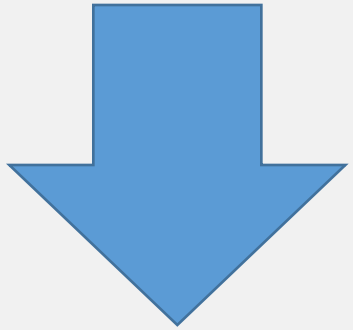
Example: Google

PaaS

Runtime environment, database, development



Google App Engine



Amazon AWS, EC2

Example: Windows Azure

- PaaS
  - Application platform in the Cloud
- Provides
  - *Compute*
    - Web, worker, and VM role
  - *Storage*
    - Blob, Table, Queue, and Azure SQL server
  - *Application fabric*
    - Service bus, access control
    - Future: cache, integration, and composite

More Cost Effective

- PaaS can be better for costs than IaaS, because systems are optimized to run applications efficiently

- IaaS may only provide hardware, and thus, clients must be in charge of load balancing and networking

Multi-tenancy

- PaaS is better suited for **multi-tenancy** because the PaaS provider optimizes its infrastructure for use by many providers

- Multi-tenancy means that many users may share the same physical computer and database

Multi-tenancy

- PaaS is better suited for multi-tenancy than an IaaS because an IaaS may (1) provide each user with his own virtual machine and (2) create a clear separation of resources

- However, in a PaaS, users may share the same machine, database, etc.

Vendor Lock-in

- PaaS may lock in applications by requiring users to develop apps using proprietary interfaces and languages
- This means that it may be difficult for users to go to another vendor to host their app
- Businesses may risk their future on the dependability of the PaaS

Development Tools

- Often, a PaaS will offer browser-based development tools
- In this way, developers can create their own applications online
- Ease of deployment: the platform takes care of the scaling for you

Principles of Software Development

- As a developer, your objective is to create an application in the quickest, most effective way possible
- You should not create applications using convoluted methods that may take a long time to complete
- The user only sees the end product, not the development process

PaaS vs. IaaS

- When you use the Cloud, remember that your decisions have long-term consequences
- If you choose to use a PaaS and get your application vendor locked in, then your business may fail if the PaaS greatly increases the vendor’s prices
- You will not be able to move to another Cloud since your app cannot be easily migrated to somewhere else

PaaS vs. IaaS

- An app that is used to fulfill a temporary need may be handled by a PaaS solution
- An app that needs to be deployed quickly may be faster developed by a PaaS
- If your software team is small, it may be better to develop a PaaS and let the PaaS provider handle the OS and networking for your team