Selected Topics in Cloud Computing

Marko Vukolić

Distributed Systems and Cloud Computing

This part of the course

- **Sample distributed systems that power clouds**
  - Amazon Dynamo
  - Apache Cassandra
  - Apache Zookeeper
  - To complement Hadoop, HDFS, Hive, RDBMSs mastered in the first part of the course

- **Cloud computing (industrial/business perspective)**

Today: Selected topics in cloud computing

- **Overview**
  - How do we define cloud computing?
  - What is the scope?
  - Some basic acronyms and terms

- **Cloud Economics (Cloudonomics) 101**

Cloud computing: beyond a buzzword

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

(FT Mar 6, 2009)

"Cloud computing achieves a quicker return on investment"

(Lindsay Armstrong of salesforce.com, Dec 2008)

" Economic downturn, the appeal of that cost advantage will be greatly magnified"

(IDC, 2008)

"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, 2008)

"No less influential than e-business"

(Gartner, 2008)

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

(Merrill Lynch, May, 2008)

Domestic cloud computing estimated to grow at 53% (moneycontrol.com, June, 2011)

Cloud computing: definition

- The "original" one, dating back to 1997

A <u>computing paradigm</u>

where the <u>boundaries</u> of computing

will be <u>determined by economic</u> rationale

rather than technical limits

Ramnath Chellapa

- Suggests very large scale
- Emphasizes the primary role of economics

Cloud computing: definition

- **NIST (US National Institute of Standards and Technology), 2011**

a model for enabling ubiquitous, convenient, on-demand network access

to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services)

that can be rapidly provisioned and released

with minimal management effort or service provider interaction

Principles really not new

Utility Computing

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

John McCarthy, 1961

Economical and convenience aspects

- **Using storage/computing without running the data/computing center yourself**
- **Much like wanting to use electricity without running a power plant at home**
- **NB: You might still install solar panels at home (see hybrid cloud later on)**

vs

Utility computing: why now?

- **Enabling technologies**
  - **Large data stores**
  - **Fiber networks**
  - **Commodity computing**
  - **Multicore machines**

+

  - **Huge data sets**
  - **Utilization/Energy**
  - **Sharing**

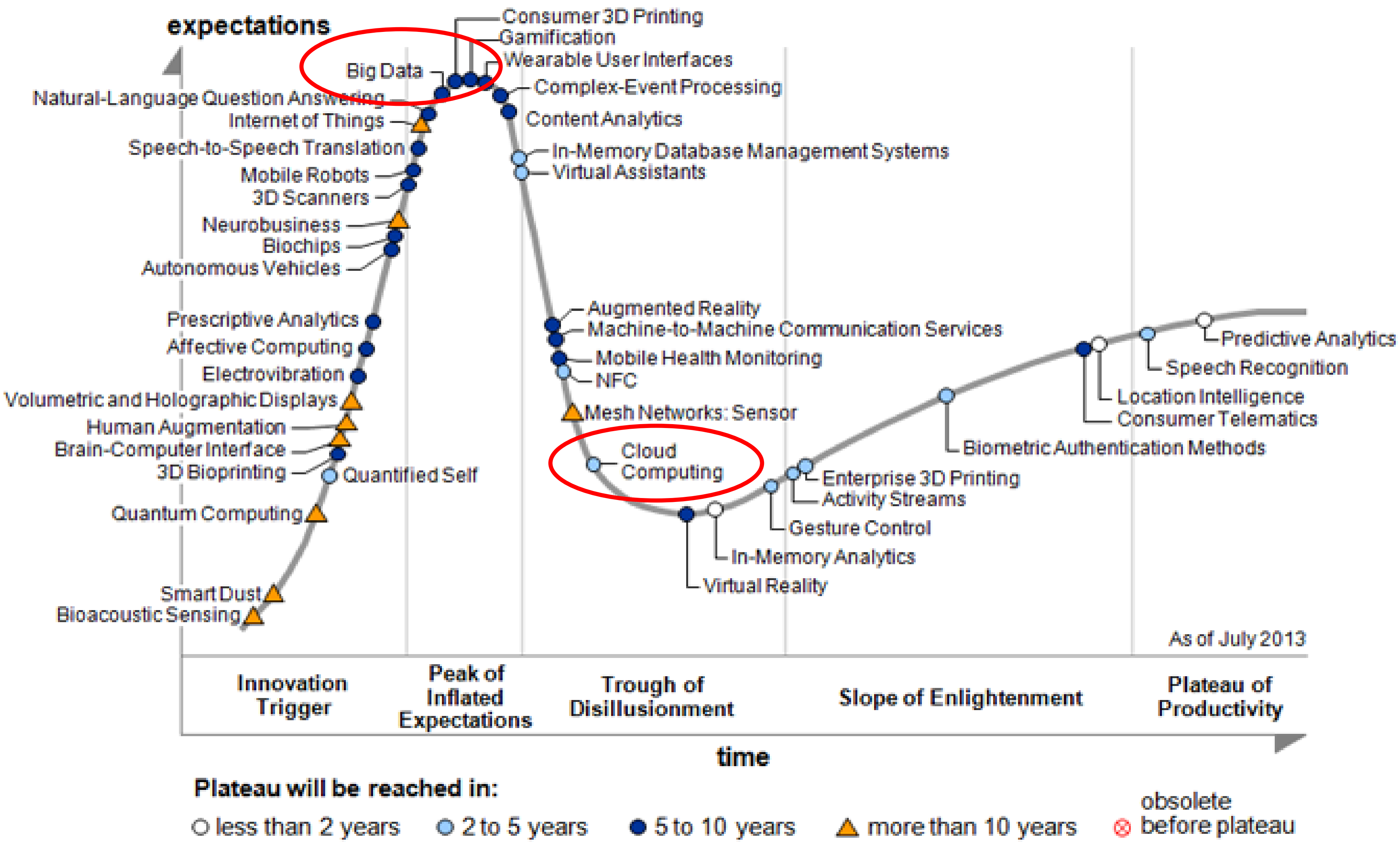
Cloud Computing: some of the keywords

- On-demand self-service
- Elasticity
- Pay-as-you-go

- Ubiquitous access

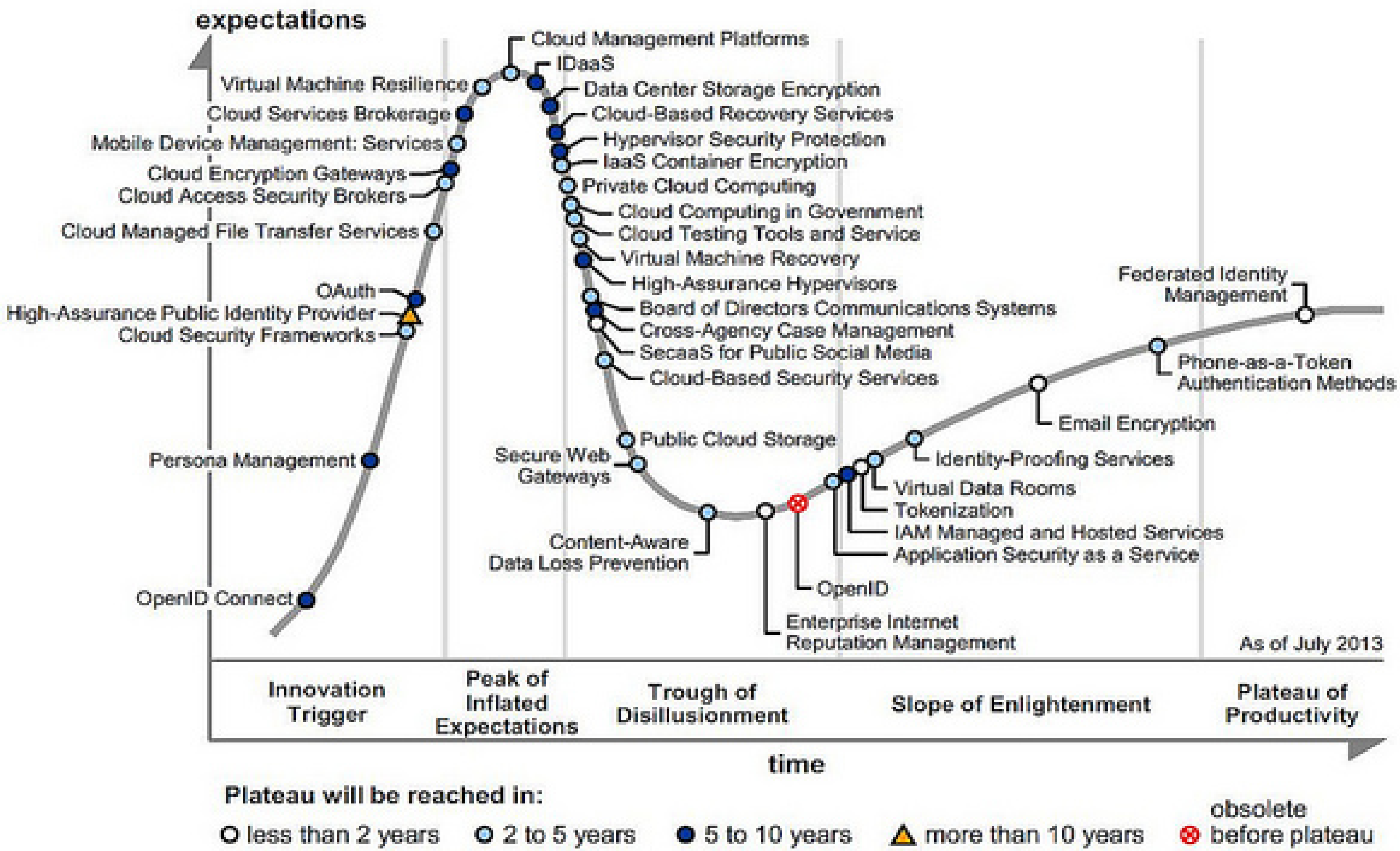
- Resource pooling / multi-tenancy
- Location opacity

Hype Cycle for Emerging Technologies



EURECOM

Cloud computing: scope and hype

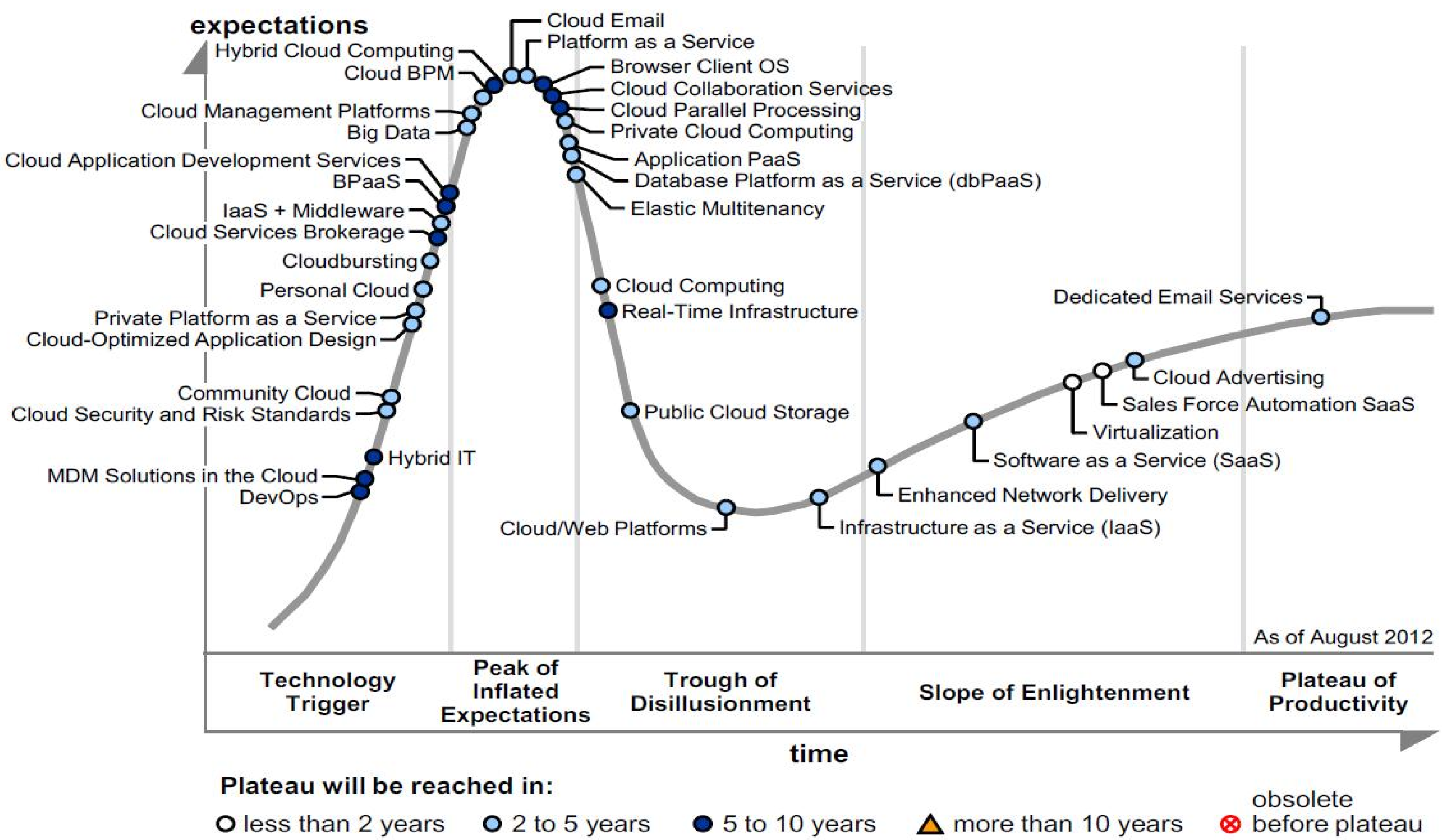


Source: Gartner (July 2013)

EURECOM

Cloud computing: scope and hype (2012)

Figure 1. Hype Cycle for Cloud Computing, 2012

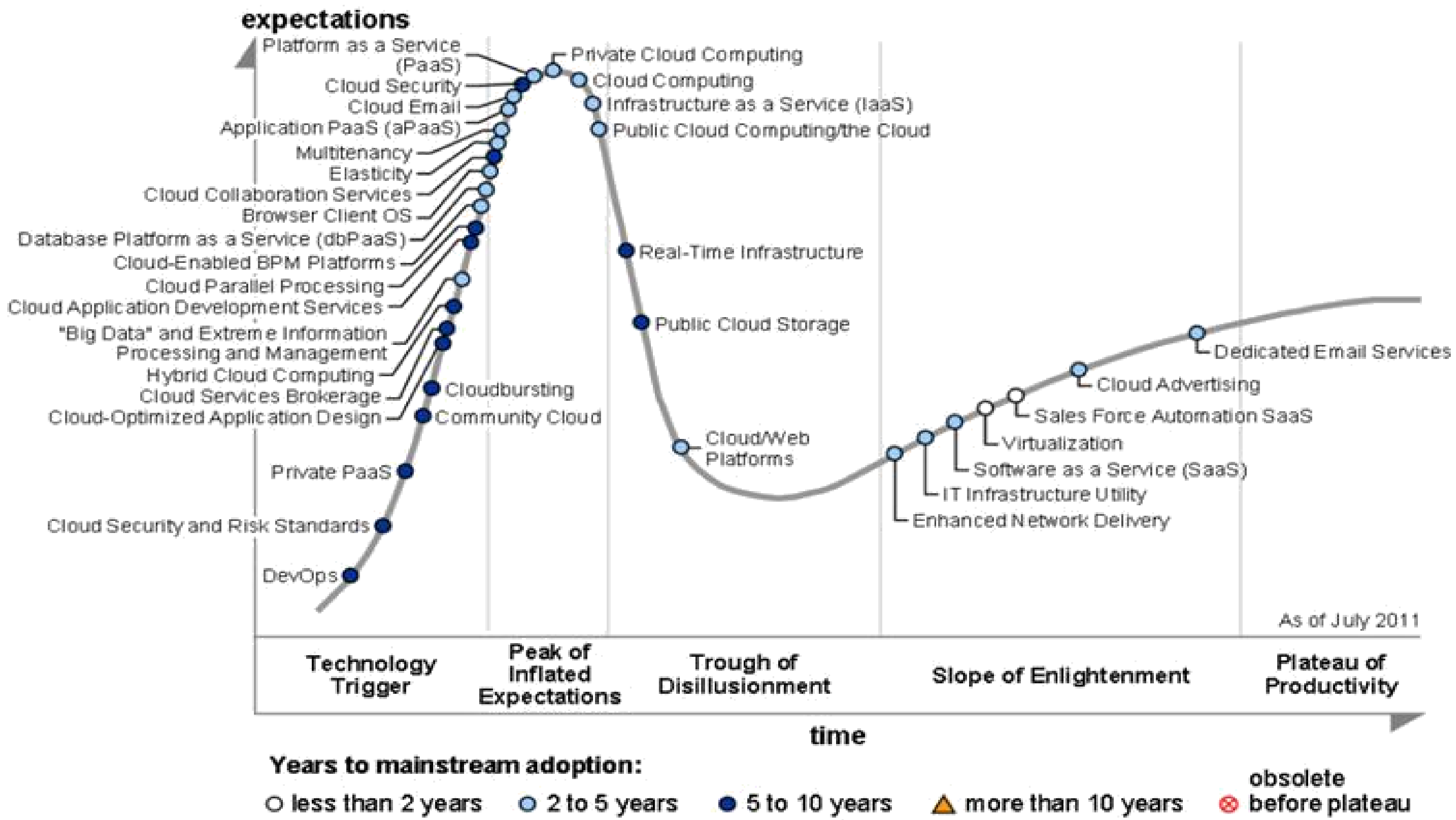


Source: Gartner (August 2012)

EURECOM

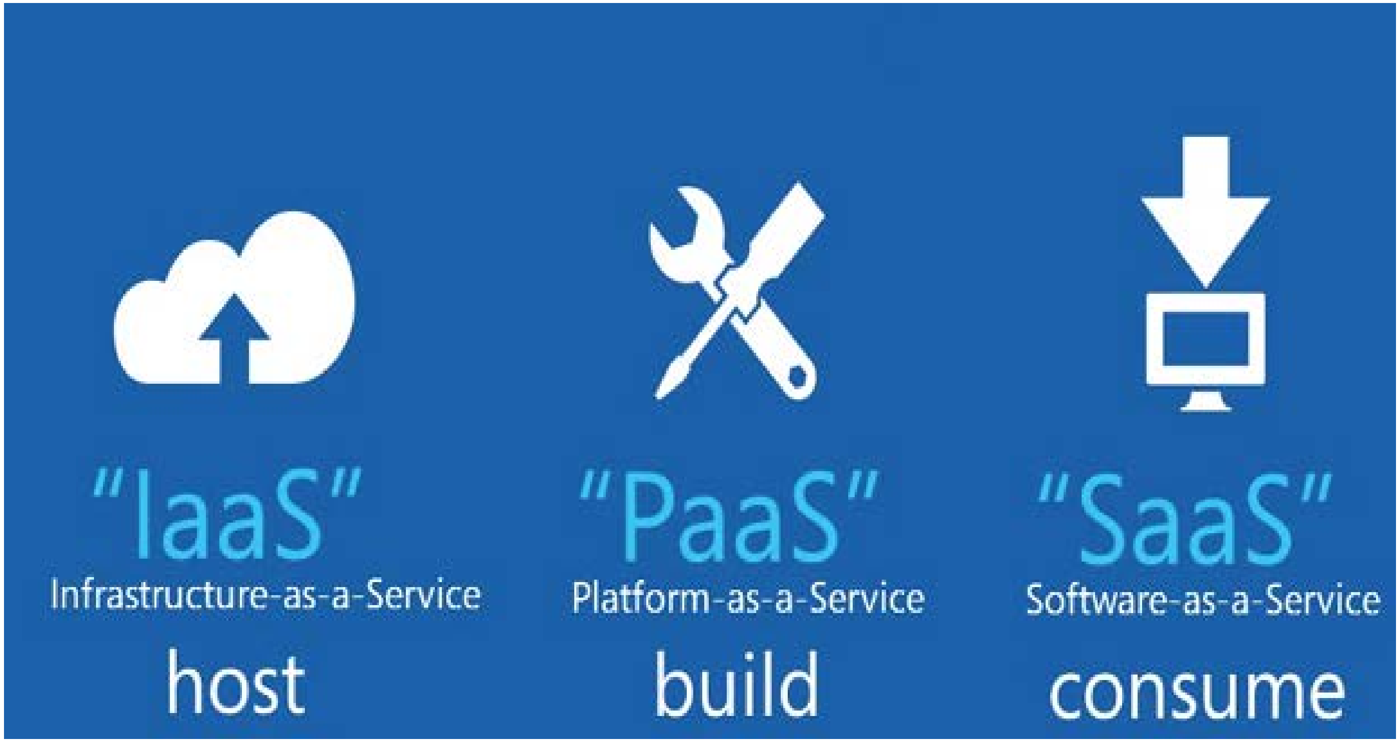
Cloud computing: scope and hype (2011)

Figure 1. Hype Cycle for Cloud Computing, 2011



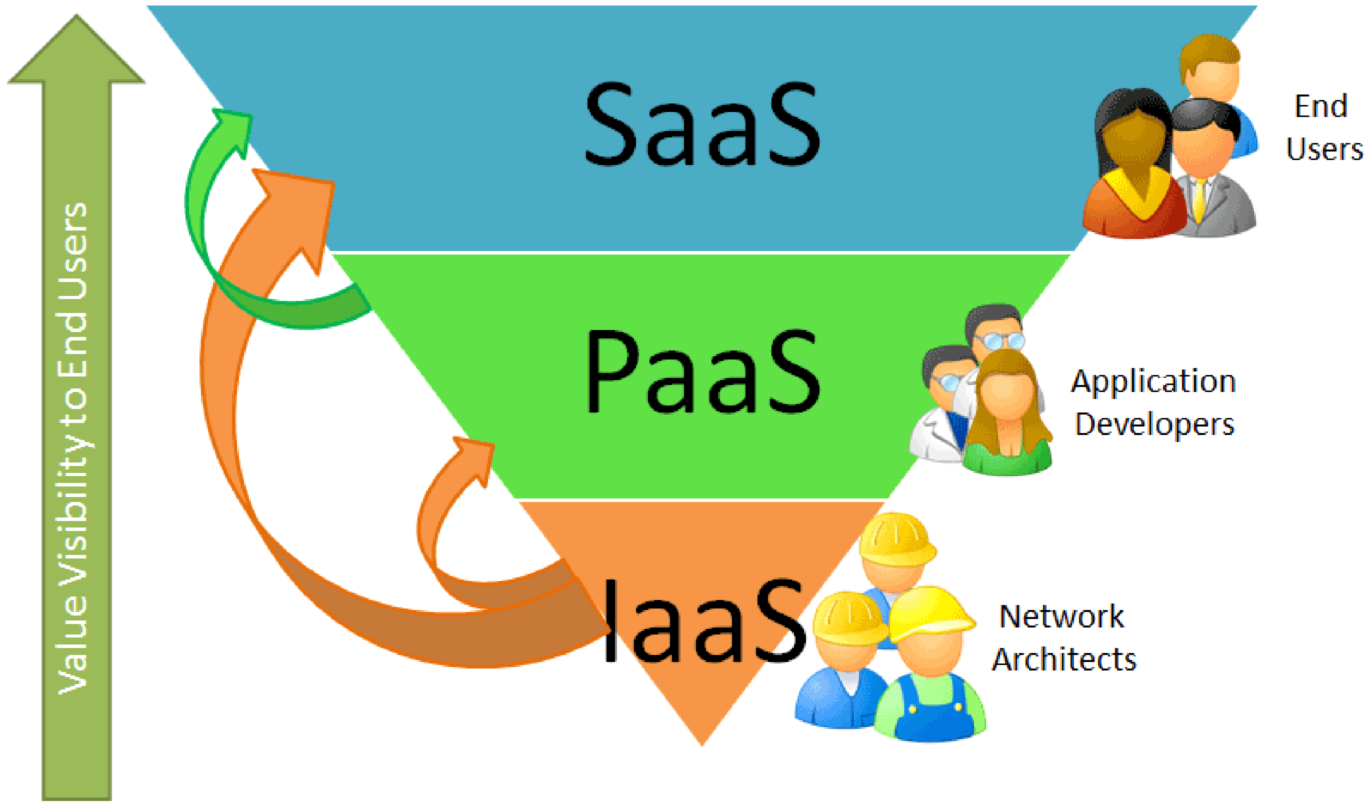
Source: Gartner (July 2011)

Cloud computing: delivery models



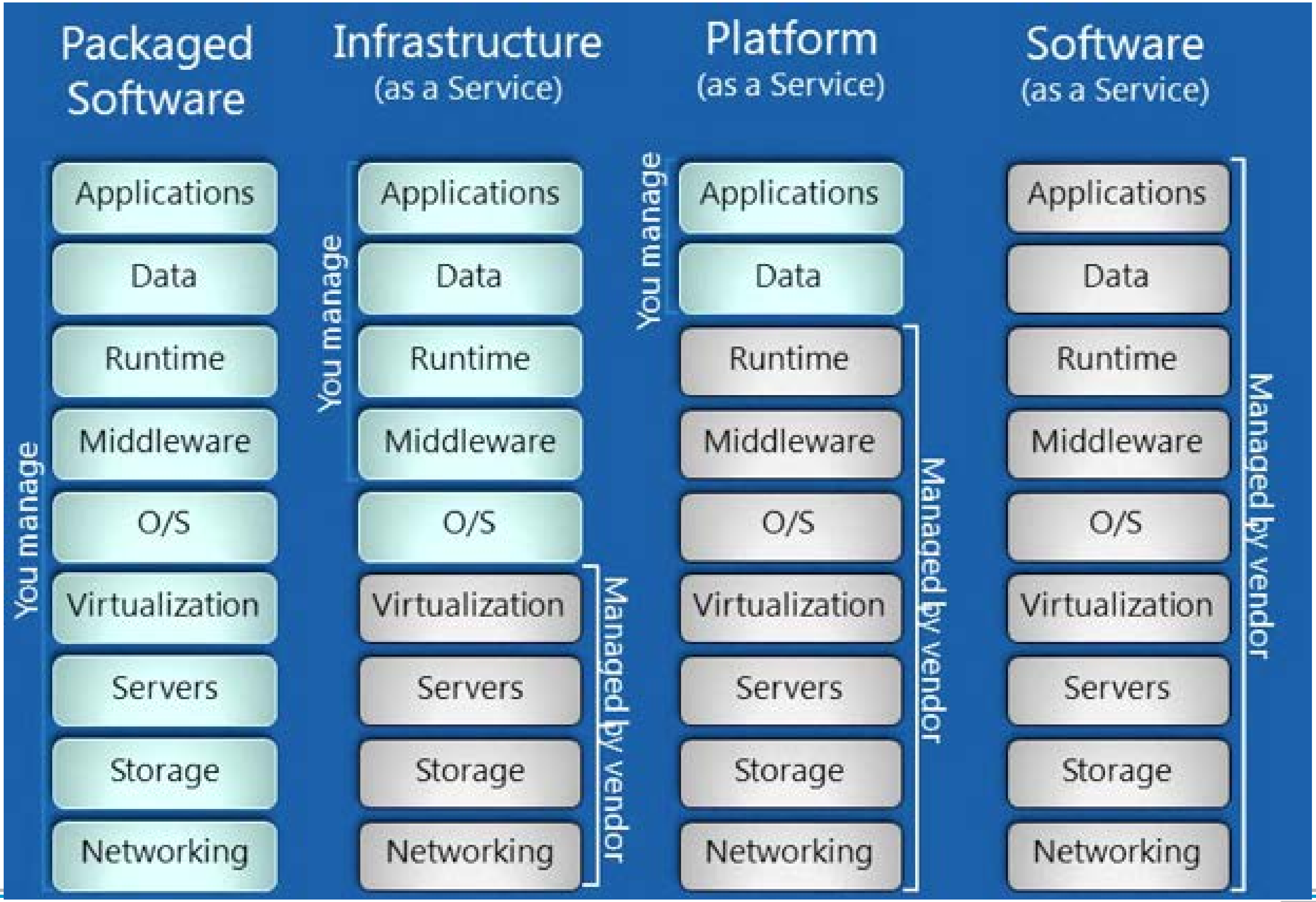
EURECOM

Cloud computing: delivery models



Network as a Service (NaaS) is becoming increasingly relevant as the 4th delivery model

Delivery models: who manages what?



EURECOM

Examples

- **SaaS**
  - Webmail, Google Apps, Dropbox, SalesForce.com
- **PaaS**
  - Windows Azure, Amazon Elastic MapReduce, Google App Engine
- **IaaS**
  - Storage / Compute
  - Amazon AWS (S3, EC2,…), Rackspace, GoGrid

Our focus in this course

- **Infrastructure as a Service**

- **Some aspects of Platform as a Service**
  - Map Reduce

Cloud Deployment Models (NIST 800-145)

- **Private cloud**
- **Community cloud**
- **Public cloud**
- **Hybrid cloud**

Private cloud

- The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers (e.g., business units).

- It may be owned, managed, and operated by the organization, a third party, or some combination of them, and it may exist on or off premises.

Community cloud

- The cloud infrastructure is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns (e.g., mission, security requirements, policy, and compliance considerations).

- It may be owned, managed, and operated by one or more of the organizations in the community, a third party, or some combination of them, and it may exist on or off premises.

Public cloud

- **The cloud infrastructure is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them.**

- **It exists on the premises of the cloud provider.**

Hybrid cloud

- **The cloud infrastructure is a composition of two or more distinct cloud infrastructures (private, community, or public).**

- **Highlight: Intercloud**
  - typically denotes a composition of two or more public clouds

Summary: cloud computing

- **Main driver: economics**
- **SaaS, PaaS, IaaS and NaaS**
  - Categorization for orientation and general idea only, sometimes the boundaries are not so clear
- **Private, Community, Public, Hybrid**
- **Affects the <u>entire</u> software (and hardware) stack!**
- **Distributed systems play the paramount role**
- **Similar to, yet different from utility, grid computing**

Today: Selected topics in cloud computing

- **Overview**
  - How do we define cloud computing?
  - What is the scope?
  - Some basic acronyms and terms

- **Cloud Economics (Cloudonomics) 101**

Cloudonomics

CLOUD → from an economic viewpoint:

1. **Common Infrastructure**
   - Resource pooling, statistical multiplexing
2. **Location opacity**
   - ubiquitous availability meeting performance requirements
   - latency reduction and user experience enhancement
3. **Online connectivity**
   - an enabler of other attributes ensuring service access
4. **Utility Pricing**
   - E.g., pay-as-you-go
5. **on-Demand Resources**
   - scalable, elastic resources provisioned and de-provisioned without delay or costs associated with change

Joe Weinman. Cloudonomics: The Business Value of Cloud Computing, Wiley, 2012.

1. Common Infrastructure

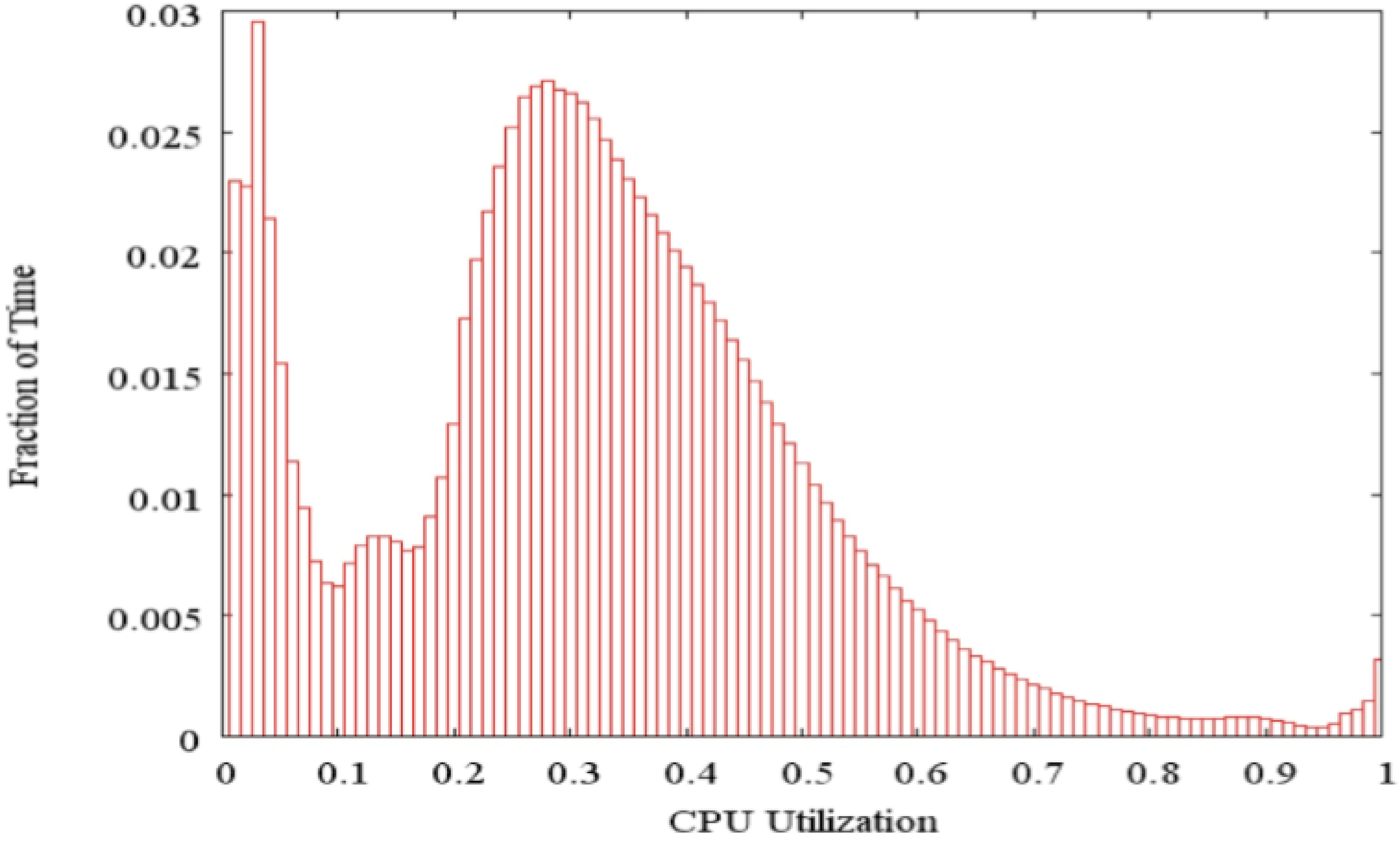
- **Resource pooling**
  - Allows economies of scale
  - Reduces overhead cost
  - Allows cloud provider more negotiating power when buying infrastructure (volume purchasing)

- **Multiplexing (multi-tenancy)**
  - Allows statistics of scale

1. Common infrastructure: Multiplexing

- **Assume you combine 2 independent infrastructures into a bigger one**
- **One is built to peak requirements**
- **The other is built to less than peak**

Google: CPU Utilization



Activity profile of a sample of 5,000 Google Servers over a period of 6 months

1. Common Infrastructure: multiplexing

- **Part of the infrastructure built to peak**
  - Load multiplexing yields higher utilization and lower cost per delivered resource wrt. unconsolidated workloads

- **For the part of the system built to less than peak**
  - Load multiplexing can reduce the unserved requests
  - Reduces a penalty function associated with such requests (e.g., a loss of revenue or a Service-Level agreement SLA violation payout).

1. Common Infrastructure: multiplexing

- **Lets define coefficient of (load) variation Cv**
  - $Cv=\sigma/|\mu|$
  - non-negative ratio of the standard deviation σ to the absolute value of the mean $|\mu|$.
  - The larger the mean for a given standard deviation, or the smaller the standard deviation for a given mean, the “smoother” the load curve is

- **Importance of *smoothness*:**
  - An infrastructure with fixed assets servicing highly variable load will achieve lower utilization than a similar one servicing relatively smooth demand.

1. Common Infrastructure: multiplexing

- **Let X1, X2…Xn be n <u>independent</u> random variables**
  - NB: might have different distributions
  - with identical standard deviation σ and positive mean μ
  - Hence, $Cv(X1)=Cv(X2)=\sigma/\mu$
- **Consider the random variable X=X1+X2+…+ Xn**
  - multiplexing
- **Statistics 101**
  - $mean(X)=mean(X1)+mean(X2)+\ldots+mean(Xn)=n\mu$
  - $var(X)=var(X1)+var(X2)+\ldots+var(Xn)=n\sigma^2$

1. Common Infrastructure: multiplexing

- **Hence standard deviation of X is**
  - stdev(X)=sqrt(Var(X))=$\sqrt{n}\,\sigma$

- **Finally Cv(X)=$\sqrt{n}\,\boldsymbol{\sigma} / n\mu = \boldsymbol{\sigma} / \sqrt{n}\mu$**
  - *i.e., Cv(X)=Cv(Xi)/$\sqrt{n}$*
  - **We obtain "smoother" aggregate load**

- **Thus, as n grows larger, the penalty function associated with insufficient or excess resources grows relatively smaller**
  - Hence, we have benefits from statistics of scale in addition to those from economies of scale

1. Common Infrastructure: multiplexing

- **Doing the maths**
- **n=100**
  - Aggregation of 100 workloads will give the 90% of the multiplexing benefit of an infinitely large cloud provider
- **n=400**
  - Aggregation of 400 workloads will give the 95% of the multiplexing benefit of an infinitely large cloud provider
- **Takeaway**
  - Midsize and private clouds might very well benefit from multiplexing statistics of scale
  - Not only giant cloud providers

1. Common Infrastructure: multiplexing

- **Mind the assumptions**
  - Independent load aggregation
- **Consider the aggregate of perfectly correlated loads**
  - Mean remains $n\mu$
  - Yet the variance is $n^2\sigma^2$
  - Hence Cv remains σ/μ (no free lunch)
- **But such a Cv remains even for infinitely large cloud providers**
  - Hence no penalty on midsize/private clouds
  - Can still profit from economies of scale

2. Location Opacity

- **Customers do not know where data is stored (where computation is performed)**
  - Intuitively, this implies <u>multiple locations for resources</u>
- **Multiple locations**
  - High availability
  - Reliability/disaster tolerance (geo-replication)
  - Performance optimizations — notably minimizing **latency**
- **New**
  - Previously users coming to terminals of computing mountains (with very fixed location), later PCs…

2. Location Opacity: Latency

- **Focal performance metric in cloud computing**
  - Throughput is important too
- **Latency largely influences design decisions in distributed systems/cloud computing**
- **What is a typical targeted latency?**
  - Depends on an application, certainly
  - Rule of the thumb: often very related to human physiology, perception and reaction times

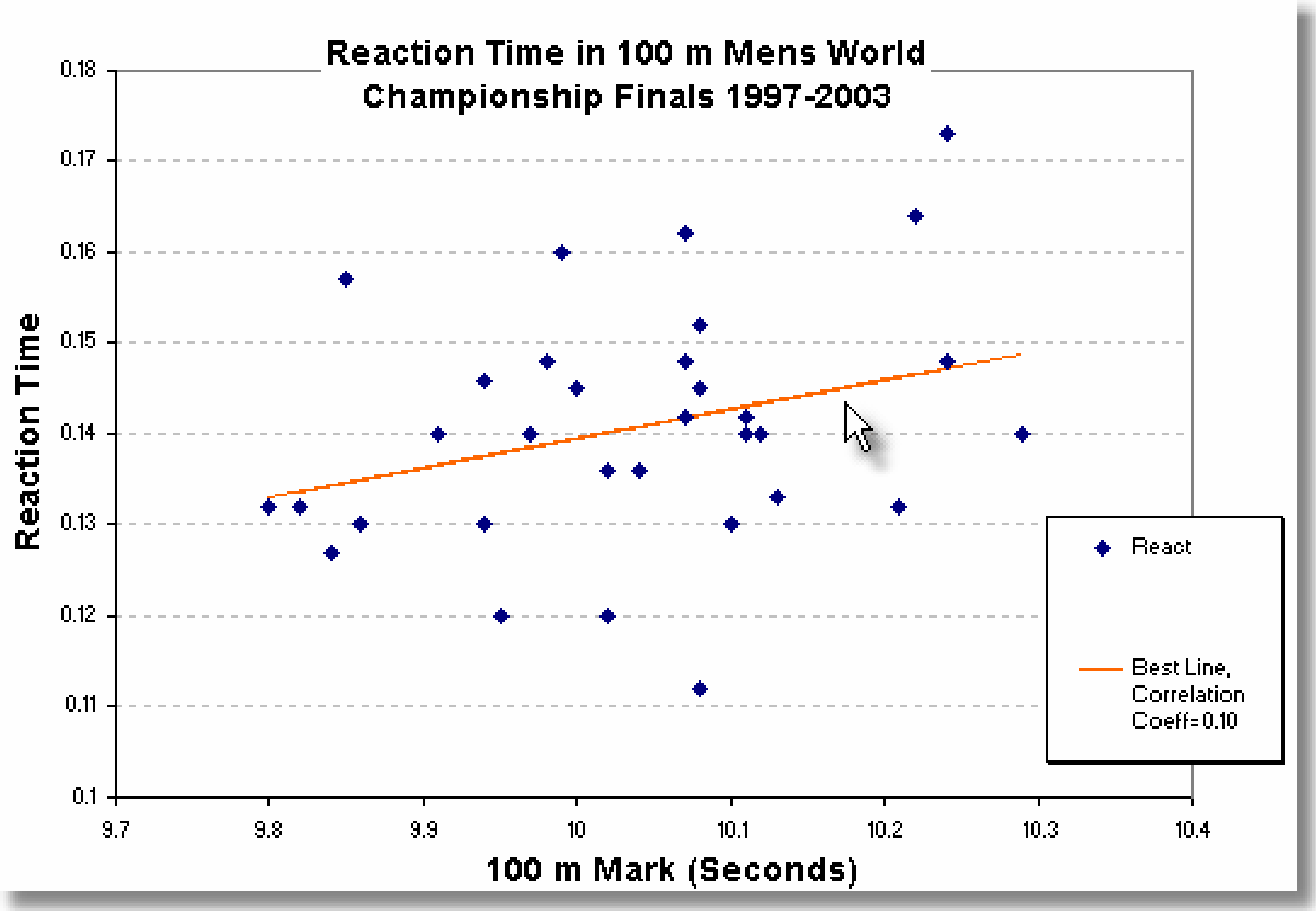
2. Location Opacity: Latency

- **Rule of the thumb ("human latency"): cca 100ms**

EURECOM

2. Location Opacity: Latency

- **Rule of the thumb ("human latency"): cca 100ms**



EURECOM

2. Location Opacity: Latency

- **Examples: VOIP, Online collaboration**
  - 200 ms inacceptable
- **Google word completion**
  - What if it took 2s?
- **Very often, single instance datacenter is not suited for these types of tasks**
  - Solution: Geo-Replication (Multiple locations for resources)

2. Location Opacity: Latency

- **Physical constraints**
  - the circumference of the Earth (cca 40000 km)
  - and the speed of light in fiber (only about 200 km/ms)
- **+ Additional latency due to, e.g., multiple roundtrips, routing, congestion, triangle inequality violations**
- **= need more than a single resource location**
  - Supporting a global user base requires a dispersed services architecture.

2. Location Opacity

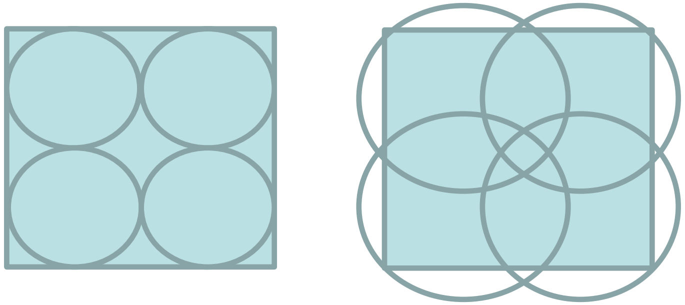
- **Need for multiple locations impact the cost**
  - (Besides, it introduces problems with consistency, partitions and consequently availability which we already discussed)

- **So how many data centers need to be deployed?**
  - Important part of the economics equations (budget)

- **Assume latency correlated with distance (albeit not perfectly)**

2. Location Opacity: Coverage

- **Subtle variations in number of nodes depending on coverage strategies**
  - circle packing vs circle covering



  - In any case the area covered is proportional to the (square of) radius and the number of service nodes

EURECOM

2. Location Opacity: Coverage

- **Planar coverage***: **the area A covered with n nodes depends on**
  - the radius r related to the latency/distance, and
  - a constant of proportionality k that depends on the packing/covering strategy
  - Thus, $A=r^2kn\pi$.
  - I.e., aiming to cover constant area, we have

$$r \sim 1/\sqrt{n}$$

- **Applies approximately to sphere coverage as well (follows from basic trigonometry)**

* **For covering smaller geo scales where Earth appears as flat**

2. Location Opacity: Coverage implications

- **Geometric reasoning yields drop in latency with $\sqrt{n}$**
- **Economic implications**
  - it doesn't take many nodes to make rapid initial gains
  - but then there are rapidly ***diminishing returns***
  - getting worst-case global network round-trip latency from 160 milliseconds to 80 or 40 or 20 takes only a several nodes, but after that, thousands or millions of nodes will only result in microsecond or nanosecond improvements

2. Location Opacity: Coverage implications

- **Economics (cnt'd)**
  - Diminishing returns make private investment difficult
- **But think (inter)cloud!**
  - What if you had only few users in a distant area?
  - Use another cloud providers' resources

3. Online connectivity

- **Much like cloud resources, clients are themselves not bound to a single location**
  - Networks providing online connectivity are ubiquitous and available
  - Wired, wireless, satellite, etc.
- **But this connectivity has costs**
  - E.g., $ per Gb transferred or the capital costs of routers or optical facilities.
- **We are skipping valuation**

4. Utility pricing

- **E.g., pay-as-you-go**

- **Q: Should you go for the public cloud if the unit CPU cycle/bit price is higher than a home-grown solution?**
  - Need to pay for Amazon’s commodity hardware
  - But also for sophisticated cooling, energy provision, smart distributed systems folks working there, Amazon profits,…
  - Might end up more expensive per CPU/data unit
  - Otherwise, it is a no-brainer to use cloud…

4. Utility pricing

- **A: It depends**
  - If cloud costs the same and the load is perfectly smooth then it is the same
  - But what if the cloud is more expensive per CPU/data unit and the load is variable?

- **Consider a car**
  - Buy (lease) for EUR 10 per day
  - vs. Rent a car for EUR 30 a day
  - If you need a car for 2 days in a month, buying would be much more costly than renting
  - **It depends on the load/demand**

4. Utility pricing

- **Turns out that in many business cases a hybrid solution is very attractive**
  - You own a daily commute car
  - But you rent a van to cover unusual demand (e.g., to move)
- **Might use public cloud to serve load spikes**
  - Christmas shopping time, slashdot effects, etc.

4. Utility Pricing: back of the envelope

- *$L(t)$*: load (demand for resources) $0<t<T$
- $P = max(L(t))$: Peak Load
- $A = Avg(L(t))$: Average Load
- *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=3*

- $C_T = \int_0^T C \times L(t)dt = A \times U \times B \times T$
- $B_T = P \times B \times T$ (since Baseline should handle peak load)
- ***When is cloud cheaper than owning?***
  - $C_T < B_T \rightarrow A \times U \times B \times T < P \times B \times T \rightarrow U < \frac{P}{A}$
  - **When Utility premium is less than Peak to Average load ratio**

5. on-Demand services

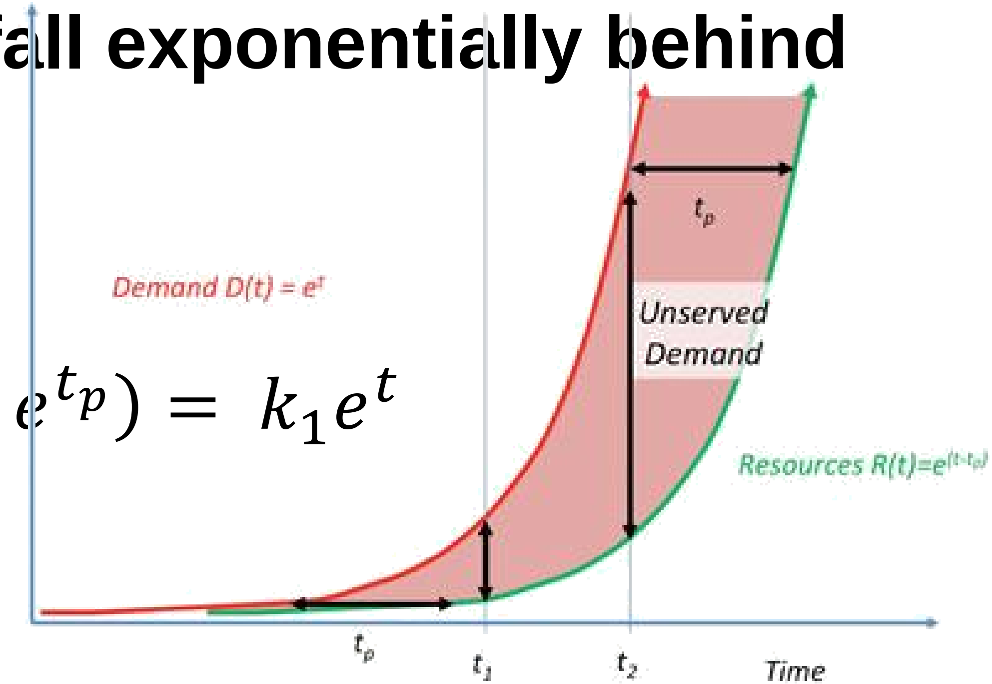
- **Owning resources can incur excessive costs**
- **Excessive resources incur costs due to**
  - weighted average cost of capital used to acquire the resources, or
  - opportunity cost of the capital not being productively employed elsewhere
  - + risk of obsolescence, premature write-offs, the risk of loss, or the cost to ensure those resources against loss
  - + floor space, and often require power and cooling
- **Insufficient resources incur lost revenue**
  - poor customer experience, loss of brand equity, etc.

5. on-Demand services: Value

- **Assume Load L(t) and owned resources R(t)**
- **You pay the penalty cost whenever R(t) does not exactly match L(t)**
- **Penalty cost P** $\sim \int |L(t) - R(t)|dt$
- **If Load is flat: P=0**
- **If Load grows linearly, steady provisioning is OK**
- **Assume now Load grows exponentially**
  - Think Big Data

5. On Demand services: Penalty Costs for exponential demand

- **P** $\sim \int |L(t) - R(t)| dt$
- **If demand is exponential (*L(t)=e^t^*), any fixed provisioning interval t_p (i.e., lag) according to the current demands will fall exponentially behind**
- ***R(t)* =** e^{t-t_p}
- ***L(t) – R(t)* =** $e^t - e^{t-t_p}$
$$= e^t(1 - e^{t_p}) = k_1 e^t$$
- **Penalty cost P=*c k*$_1$*e*^{t}**
- ***Cloud:*** $t_p \rightarrow 0 \Rightarrow$ **P=0**



Exponential Growth with Continuous Monitoring And Non-Zero Provisioning Interval

Other aspects: Behavioral Cloudonomics

- **Human decisions are not always purely inspired by maximizing mathematical functions**
  - See “Allais paradox”
- **Pros:**
  - attraction of “free” offers
  - The lack of upfront investment in using public clouds is extremely attractive
- **Cons:**
  - customers may recognize the financial advantage of pay-as-you-go, but avoid it due to a “flat-rate” bias
    - E.g. fear of an unexpected large monthly cell phone bill favoring flat-rate

Putting things together?

- **Complexity is often intractable**
- **Satisfying variable load with constraints (e.g. distance) is computationally intractable**
  - Cloud computing load/demand satisfiability is NP-complete
- **Even with exactly right aggregate capacity in a cloud, it may be intractable to find the right assignment of capacity to demand**
  - E.g. Hadoop map job scheduling wrt File chunk locations in HDFS
- **Common Infrastructure and Location Opacity (latency optimization) are usually a tradeoff**

Further reading

J. Weinman. *Cloudonomics: The Business Value of Cloud Computing*, Wiley, 2012

L.A. Barroso, Jimmy Clidaras and U. Hölzle. The Datacenter as a Computer: *An Introduction to the Design of Warehouse-Scale Machines,* Morgan&Claypool, 2nd ed. July 2013

http://www.morganclaypool.com/doi/abs/10.2200/S00516ED2V01Y201306CAC024

(accessible from EURECOM network)

Exercise: Read/Write locks

WriteLock(filename)

```
1:      myLock=create(filename + “/write-”, “”, EPHEMERAL & SEQUENTIAL)
2:      C = getChildren(filename, false)
3:      if myLock is the lowest znode in C then return
4:      else
5:              precLock = znode in C ordered just before myLock
6:              if exists(precLock, true)
7:                      wait for precLock watch
8:              goto 2:
```

Exercise: Read/Write Locks

ReadLock(filename)

```
1:    myLock=create(filename + "/read-", "", EPHEMERAL & SEQUENTIAL)
2:    C = getChildren(filename, false)
3:    if no "/write-" znode in C then return
4:    else
5:        precLock = "/write-" znode in C ordered just before myLock
6:        if exists(precLock, true)
7:            wait for precLock watch
8:        goto 2:
```

Release(filename)

```
      delete(myLock)
```