IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

NPTEL

CLOUD COMPUTING

Introduction to DOCKER Container

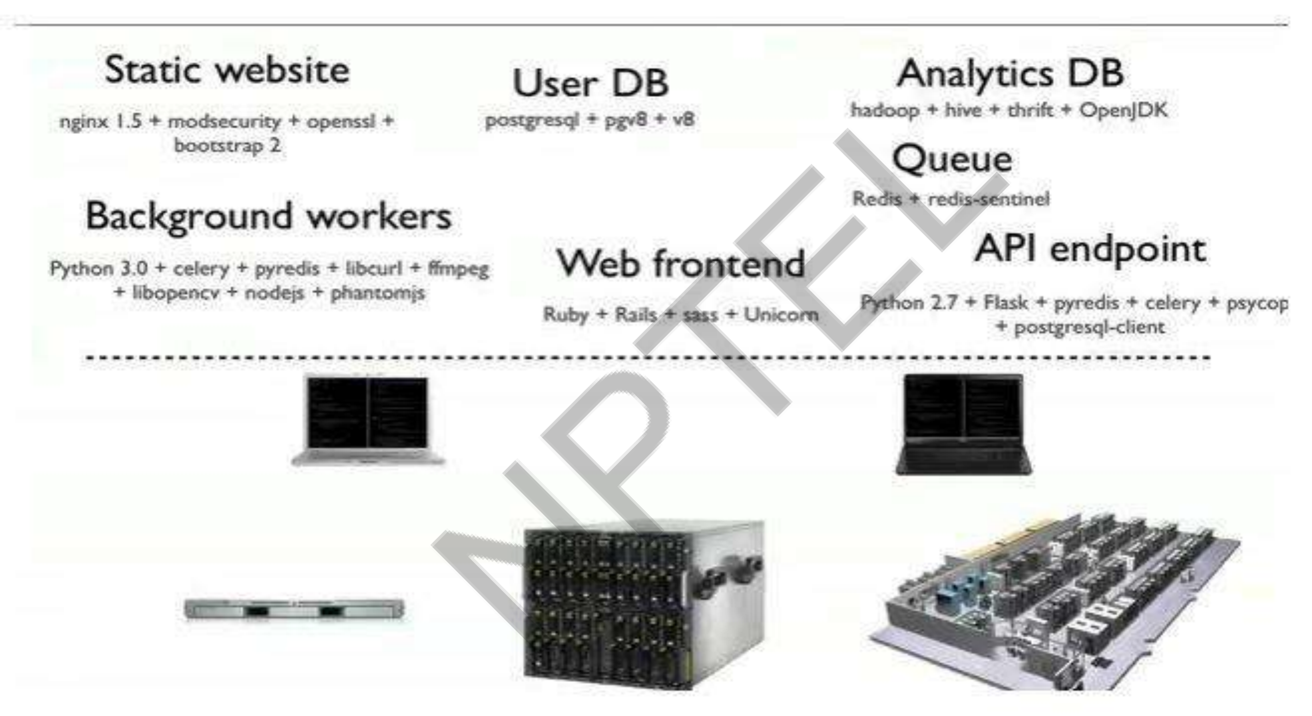
PROF. SOUMYA K. GHOSH
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
IIT KHARAGPUR

Docker

- Docker is a container management service (initial release: March 2013)
- Main features of Docker are *develop*, *ship* and *run anywhere*.
- Docker aims at facilitating developers to easily develop applications, ship them into containers which can then be deployed anywhere.
- It has become the buzzword for modern world development, especially in the face of Agile-based projects.

Ref: https://www.tutorialspoint.com/docker/

Infrastructure and Software Stack



Static website

nginx 1.5 + modsecurity + openssl + bootstrap 2

User DB

postgresql + pgv8 + v8

Analytics DB

hadoop + hive + thrift + OpenJDK

Queue

Redis + redis-sentinel

Background workers

Python 3.0 + celery + pyredis + libcurl + ffmpeg + libopencv + nodejs + phantomjs

Web frontend

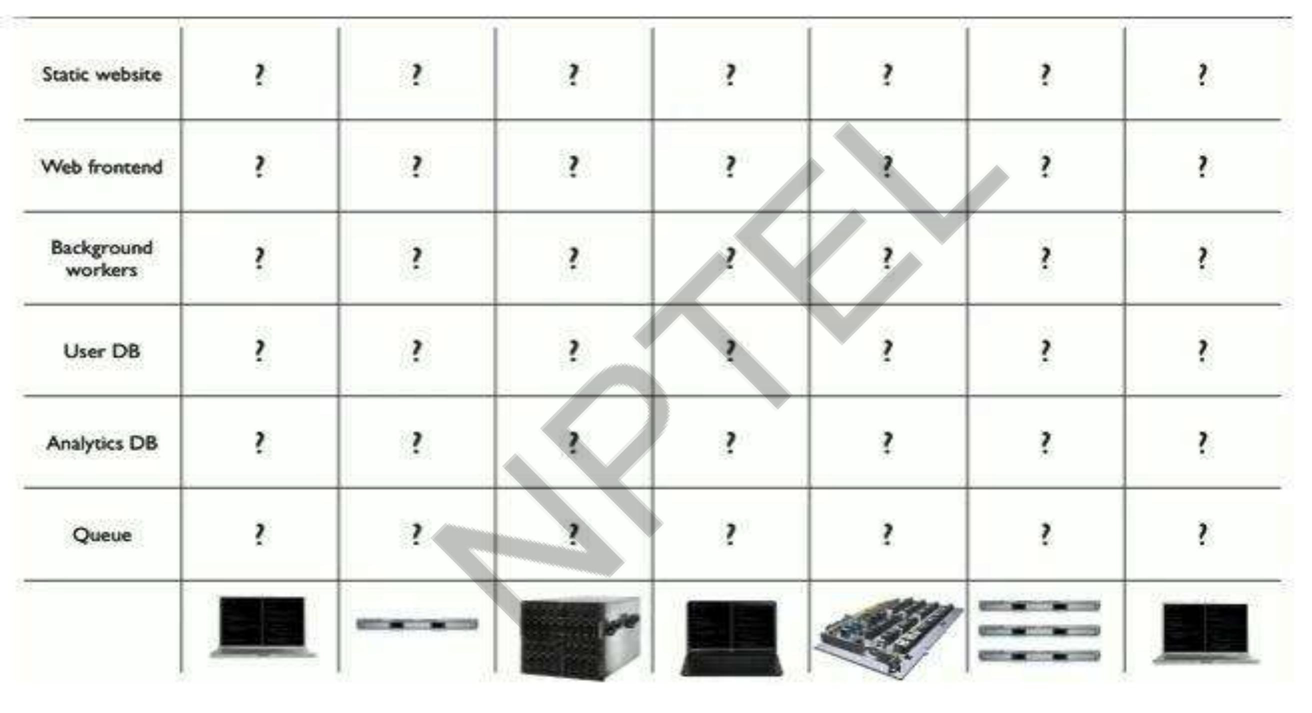
Ruby + Rails + sass + Unicorn

API endpoint

Python 2.7 + Flask + pyredis + celery + psycop + postgresql-client

Ref: Internet/YouTube

Goal: Interoperability



Static website	?	?	?	?	?	?	?
Web frontend	?	?	?	?	?	?	?
Background workers	?	?	?	?	?	?	?
User DB	?	?	?	?	?	?	?
Analytics DB	?	?	?	?	?	?	?
Queue	?	?	?	?	?	?	?

Ref: Internet/YouTube

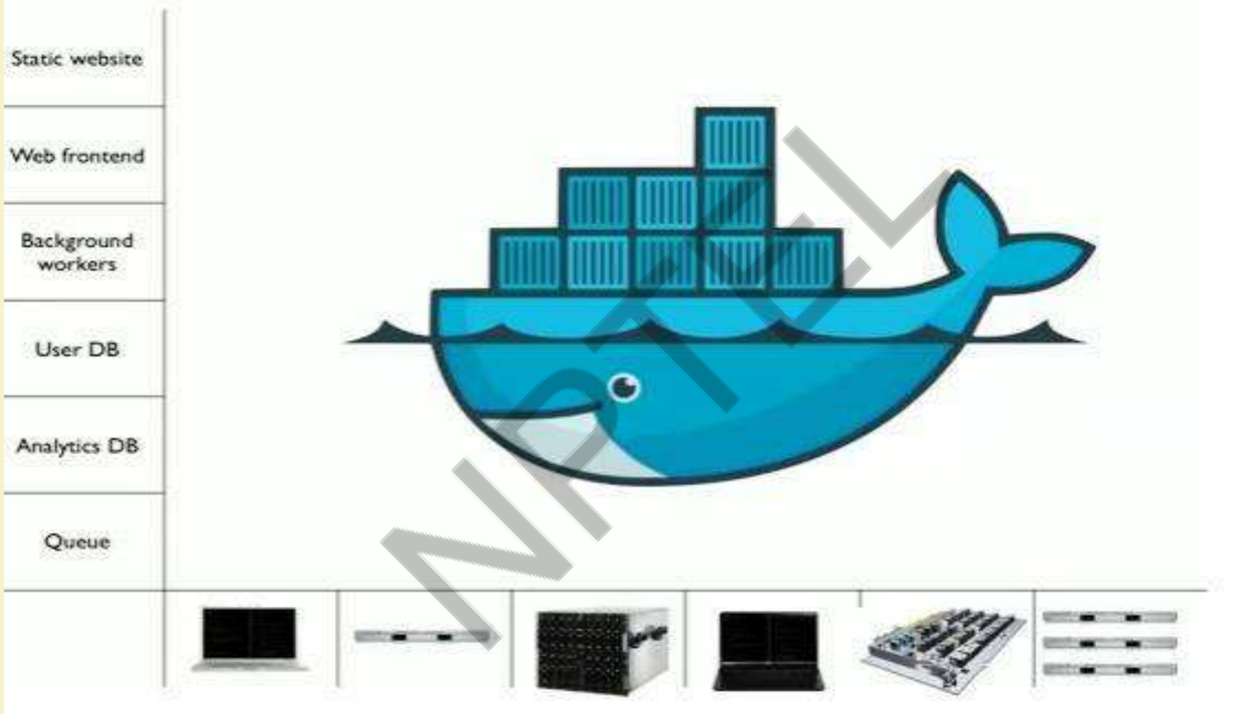
“Shipping”

Ref: Internet/YouTube

“Shipping”

Ref: Internet/YouTube

“Docker”



Ref: Internet/YouTube

Docker – Features

- Docker has the ability to reduce the size of development by providing a smaller footprint of the operating system via containers.
- With containers, it becomes easier for software teams, such as development, QA and Operations to work seamlessly across applications.
- One can deploy Docker containers anywhere, on any physical and virtual machines and even on the cloud.
- Since Docker containers are pretty lightweight, they are very easily scalable.

Ref: https://www.tutorialspoint.com/docker/

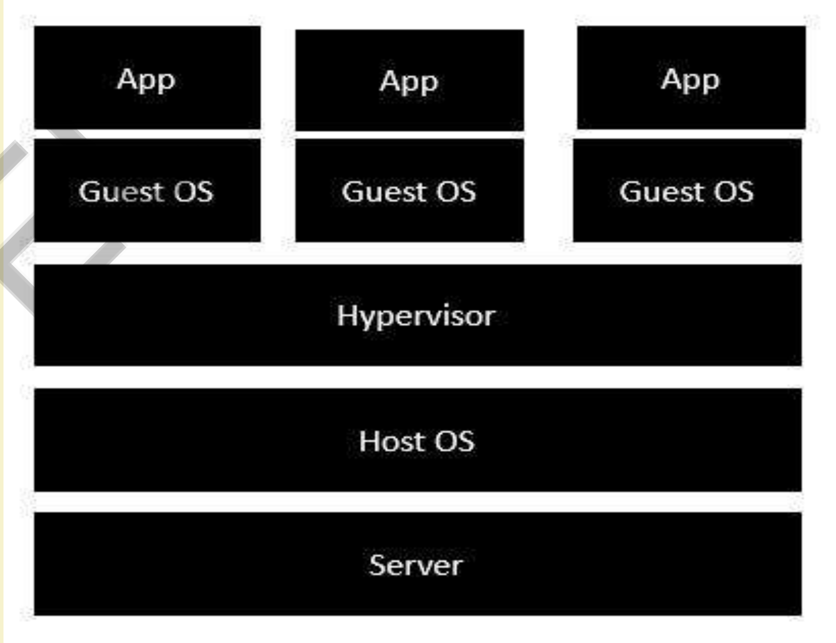
Docker – Components

- Docker for Mac − It allows one to run Docker containers on the Mac OS.
- Docker for Linux − It allows one to run Docker containers on the Linux OS.
- Docker for Windows − It allows one to run Docker containers on the Windows OS.
- Docker Engine − It is used for building Docker images and creating Docker containers.
- Docker Hub − This is the registry which is used to host various Docker images.
- Docker Compose − This is used to define applications using multiple Docker containers.

Ref: https://www.tutorialspoint.com/docker/

Traditional Virtualization

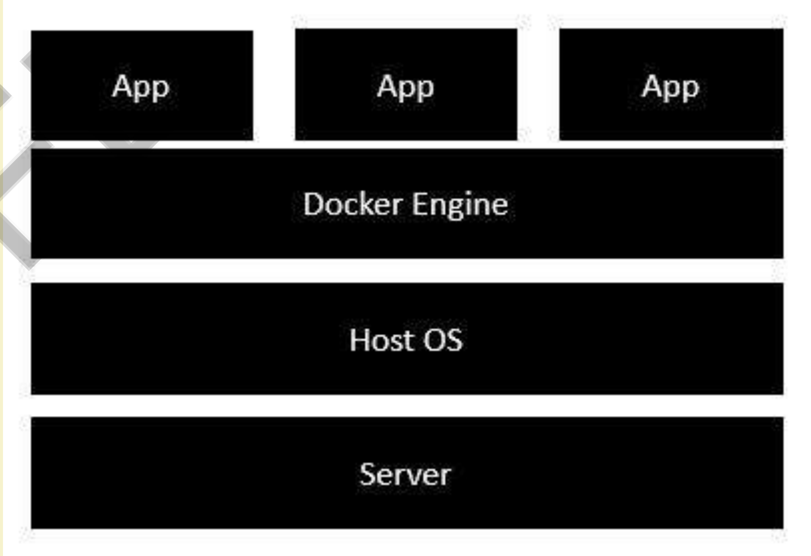
- Server is the physical server that is used to host multiple virtual machines.
- Host OS is the base machine such as Linux or Windows.
- Hypervisor is either VMWare or Windows Hyper V that is used to host virtual machines.
- One would then install multiple operating systems as virtual machines on top of the existing hypervisor as Guest OS.
- One would then host your applications on top of each Guest OS.



Ref: https://www.tutorialspoint.com/docker/

Docker – Architecture

- Server is the physical server that is used to host multiple virtual machines.
- Host OS is the base machine such as Linux or Windows.
- Docker engine is used to run the operating system which earlier used to be virtual machines as Docker containers.
- All of the Apps now run as Docker containers.



Ref: *https://www.tutorialspoint.com/docker/*

Container?

- Containers are an abstraction at the app layer that packages code and dependencies together.
- Multiple containers can run on the same machine and share the OS kernel with other containers, each running as isolated processes in user space.
- Containers take up less space than VMs (container images are typically tens of MBs in size), and start almost instantly.

Ref: https://www.docker.com/

Container (contd...)

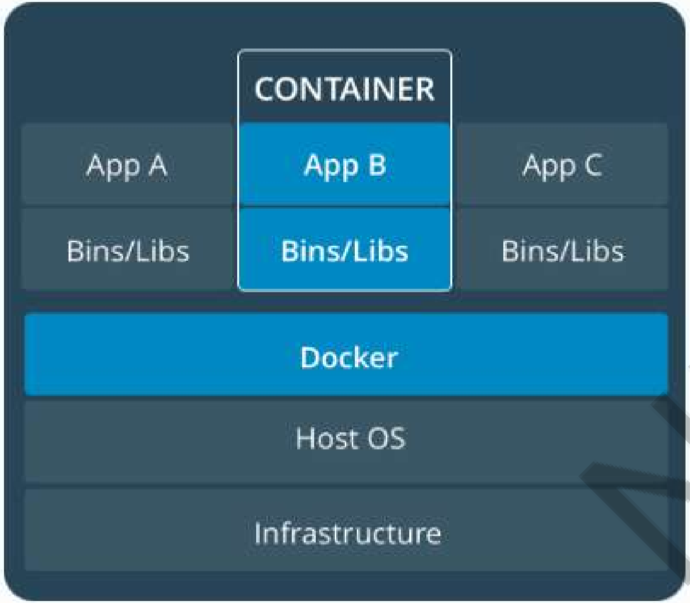
- An ***image*** is a lightweight, stand-alone, executable package that includes everything needed to run a piece of software, including the code, a runtime, libraries, environment variables, and config files.

- A ***container*** is a runtime instance of an image—what the image becomes in memory when actually executed. It runs completely isolated from the host environment by default, only accessing host files and ports if configured to do so.

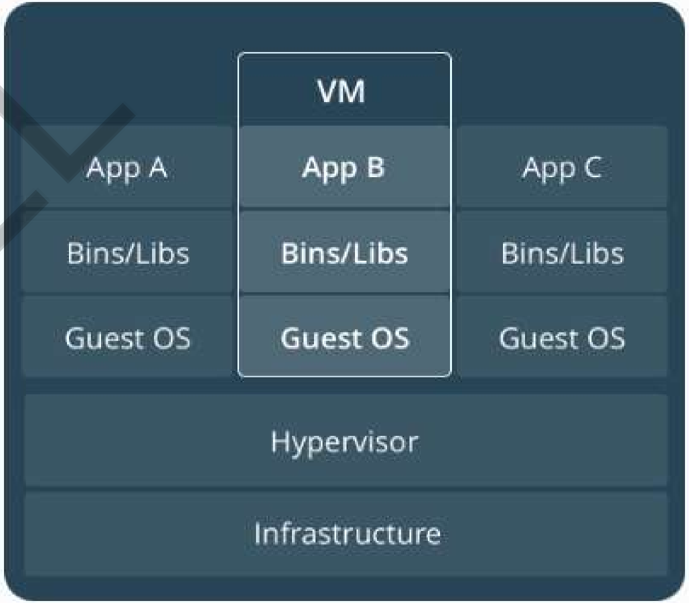
- Containers run apps natively on the host machine's kernel. They have better performance characteristics than virtual machines that only get virtual access to host resources through a hypervisor. Containers can get native access, each one running in a discrete process, taking no more memory than any other executable.

Ref: https://www.docker.com/

Containers and Virtual Machines



Container



VM

Ref: https://www.docker.com/

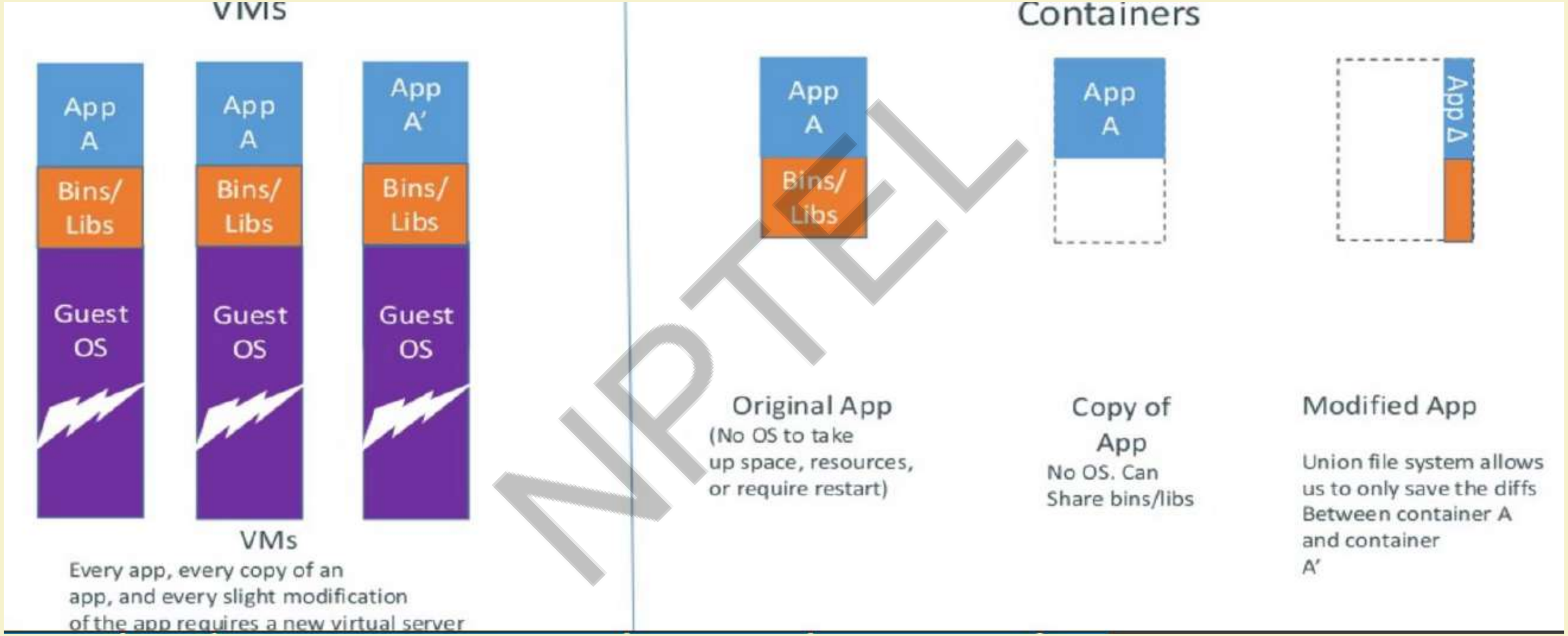
Virtual Machines and Containers

- **Virtual machines** run guest operating systems - the OS layer in each box.
- Resource intensive, and the resulting disk image and application state is an entanglement of OS settings, system-installed dependencies, OS security patches, and other easy-to-lose, hard-to-replicate ephemera.

- **Containers** can share a single kernel, and the only information that needs to be in a container image is the executable and its package dependencies, which never need to be installed on the host system.
- These processes run like native processes, and can be managed individually
- Because they contain all their dependencies, there is no configuration entanglement; a containerized app “runs anywhere”

Ref: https://www.docker.com/

Docker containers are lightweight



Every app, every copy of an app, and every slight modification of the app requires a new virtual server

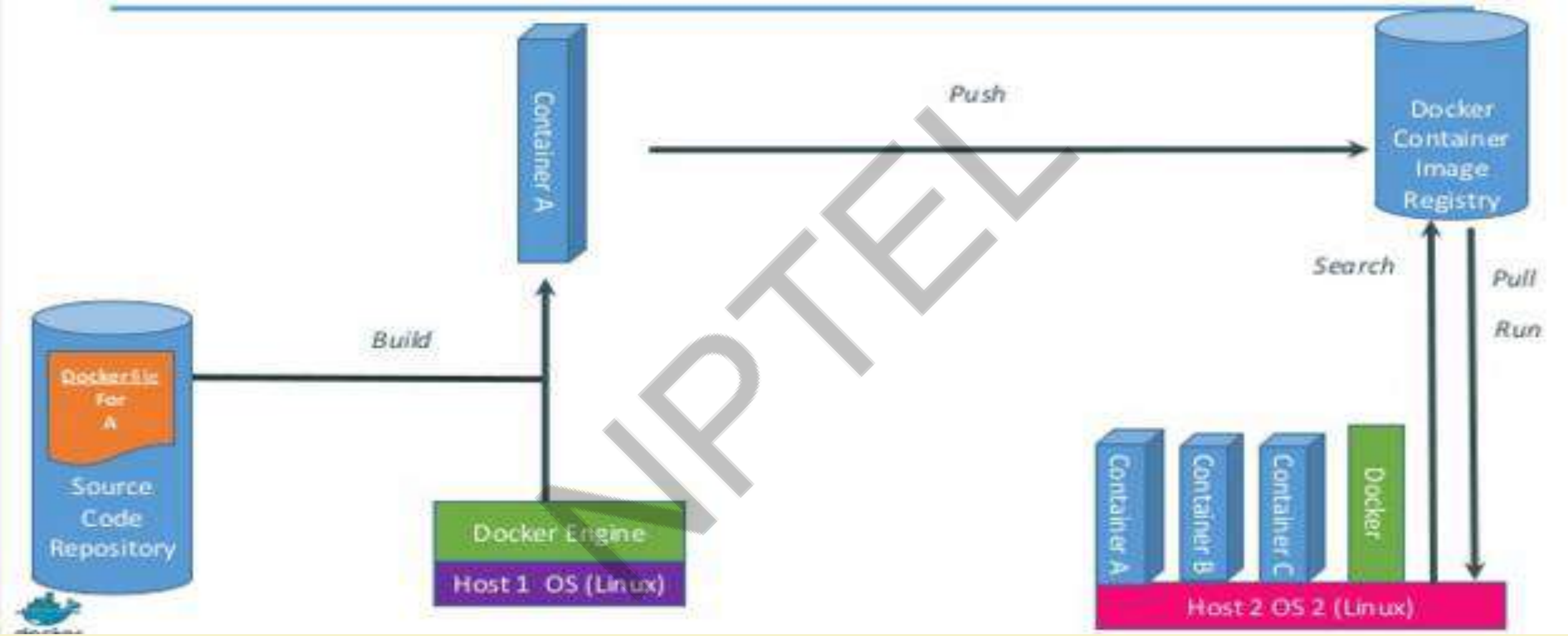
(No OS to take up space, resources, or require restart)

No OS. Can Share bins/libs

Union file system allows us to only save the diffs Between container A and container A′

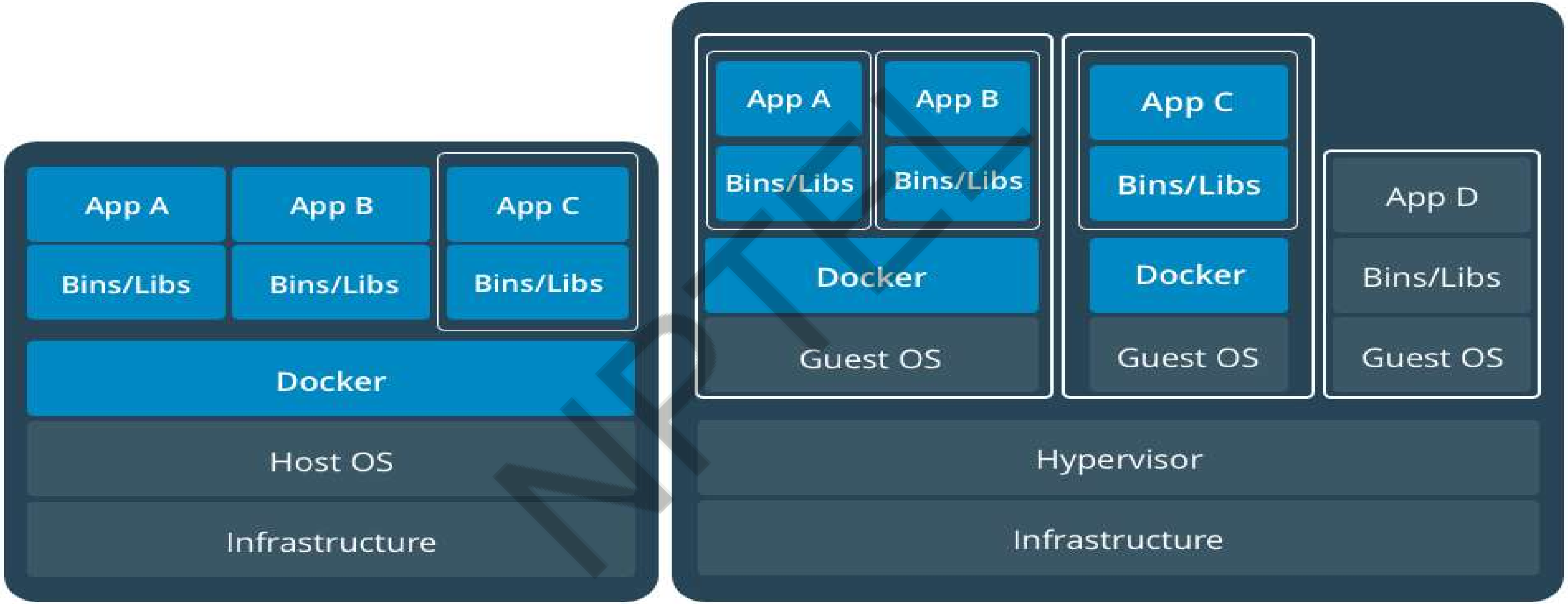
Ref: Internet

How does Docker work



Source: Internet

Containers and Virtual Machines Together



Ref: https://www.docker.com/

Why is Docker needed for applications?

- Application level virtualization.
- A single host can run several spatial applications for utilization of resources.
- Build once, deploy anywhere, run anywhere.
- Better collaboration while development of applications.

Ref: https://www.docker.com/

Terminology - Image

- Persisted snapshot that can be run
  - *images:* List all local images
  - *run*: Create a container from an image and execute a command in it
  - *tag*: Tag an image
  - *pull*: Download image from repository
  - *rmi*: Delete a local image
    - This will also remove intermediate images if no longer used

Ref: https://www.docker.com/

Terminology - Container

- Runnable instance of an image
  - *ps:* List all running containers
  - *ps –a*: List all containers (incl. stopped)
  - *top*: Display processes of a container
  - *start*: Start a stopped container
  - *stop*: Stop a running container
  - *pause*: Pause all processes within a container
  - *rm*: Delete a container
  - *commit*: Create an image from a container

Ref: https://www.docker.com/

Dockerfile

- Create images automatically using a build script: «Dockerfile»
- Can be versioned in a version control system like Git or SVN, along with all dependencies
- Docker Hub can automatically build images based on dockerfiles on Github

Ref: https://www.docker.com/

Docker Hub

- Public repository of Docker images
  - https://hub.docker.com/
- Automated: Has been automatically built from Dockerfile
  - Source for build is available on GitHub

Ref: https://www.docker.com/

Docker – Usage

- Docker is the world’s leading software container platform.
- Developers use Docker to eliminate “works on my machine” problems when collaborating on code with co-workers.
- Operators use Docker to run and manage apps side-by-side in isolated containers to get better compute density.
- Enterprises use Docker to build agile software delivery pipelines to ship new features faster, more securely and with confidence for both Linux, Windows Server, and Linux-on-mainframe apps.

Ref: https://www.docker.com/

Thank You!

IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

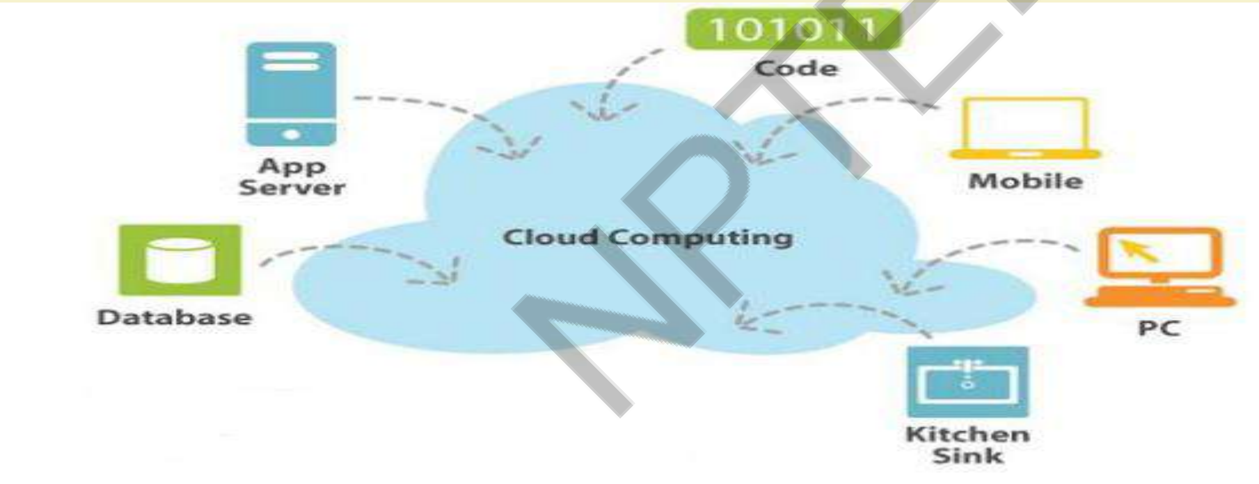
CLOUD COMPUTING

Green Cloud

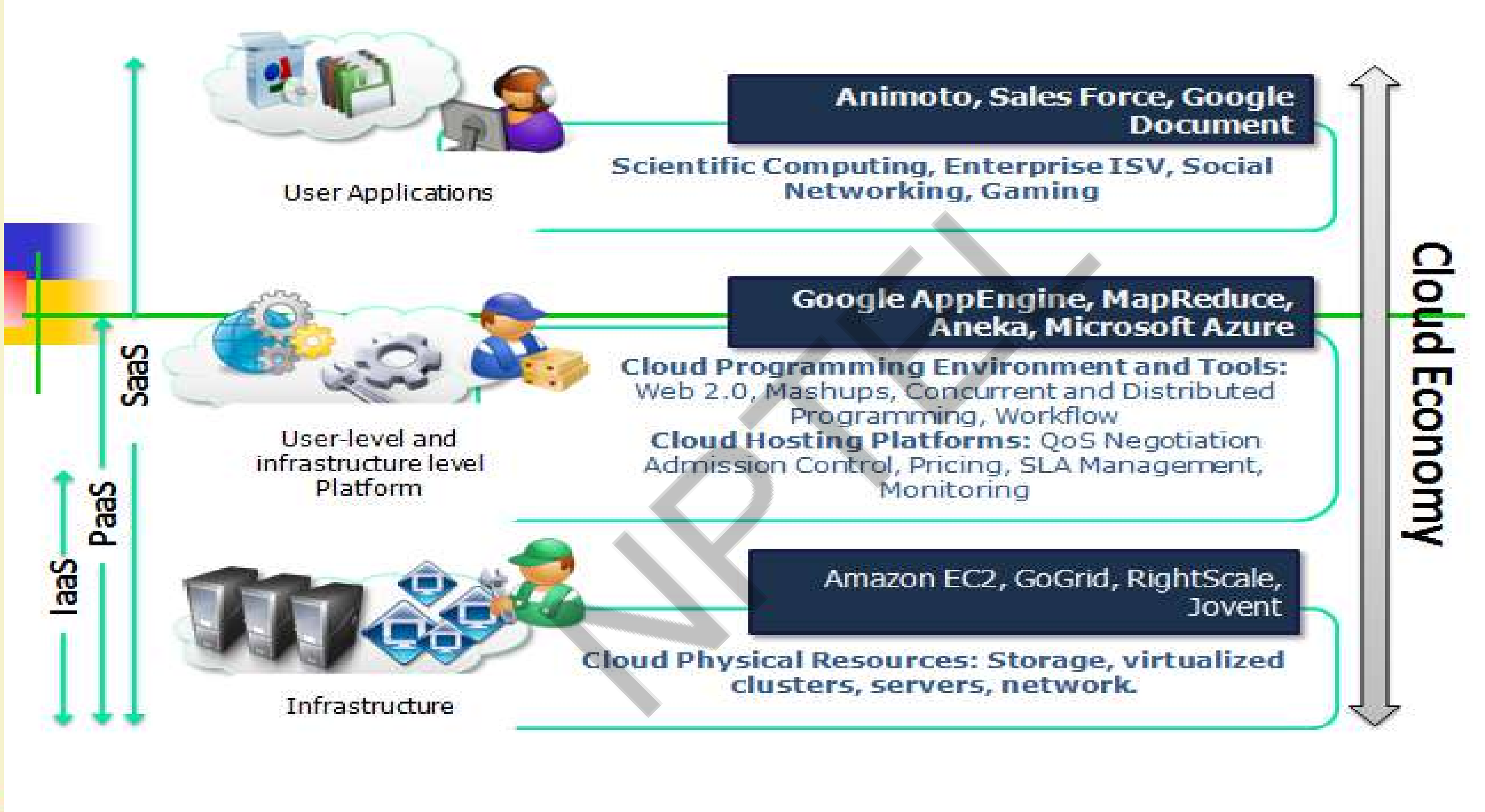
PROF. SOUMYA K. GHOSH
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
IIT KHARAGPUR

Cloud Computing

Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources like networks, servers, storage, applications, and services.



Source: Internet



User Applications

Animoto, Sales Force, Google Document

Scientific Computing, Enterprise ISV, Social Networking, Gaming

User-level and infrastructure level Platform

Google AppEngine, MapReduce, Aneka, Microsoft Azure

Cloud Programming Environment and Tools: Web 2.0, Mashups, Concurrent and Distributed Programming, Workflow

Cloud Hosting Platforms: QoS Negotiation Admission Control, Pricing, SLA Management, Monitoring

Infrastructure

Amazon EC2, GoGrid, RightScale, Jovent

Cloud Physical Resources: Storage, virtualized clusters, servers, network.

Source: Internet

Green Cloud ?

- Green computing is the environmentally responsible and eco-friendly use of computers and their resources.
- In broader terms, it is also defined as the study of designing, manufacturing or engineering, using and disposing of computing devices in a way that reduces their environmental impact.
- Green Cloud computing is envisioned to achieve not only efficient processing and utilization of computing infrastructure, but also minimize energy consumption.

Source: Internet

Cloud Advantages

- **Reduce spending on technology infrastructure.** Maintain easy access to information with minimal upfront spending. Pay as you go based on demand.
- **Globalize your workforce on the cheap**. People worldwide can access the cloud, provided they have an Internet connection.
- **Streamline processes**. Get more work done in less time with less people.
- **Reduce capital costs**. There's no need to spend big money on hardware, software or licensing fees.
- **Improve accessibility**. You have access anytime, anywhere, making your life so much easier!
- **Minimize licensing new software**. Stretch and grow without the need to buy expensive software licenses or programs.
- **Improve flexibility**. You can change direction without serious financial issues at stake.

Source: Internet

Cloud – Challenge

- Gartner Report 2007: IT industry contributes 2% of world's total CO2 emissions
- U.S. EPA Report 2007: 1.5% of total U.S. power consumption used by data centers which has more than doubled since 2000 and costs $4.5 billion

>> **Need of Green Cloud Computing….**

Source: Internet

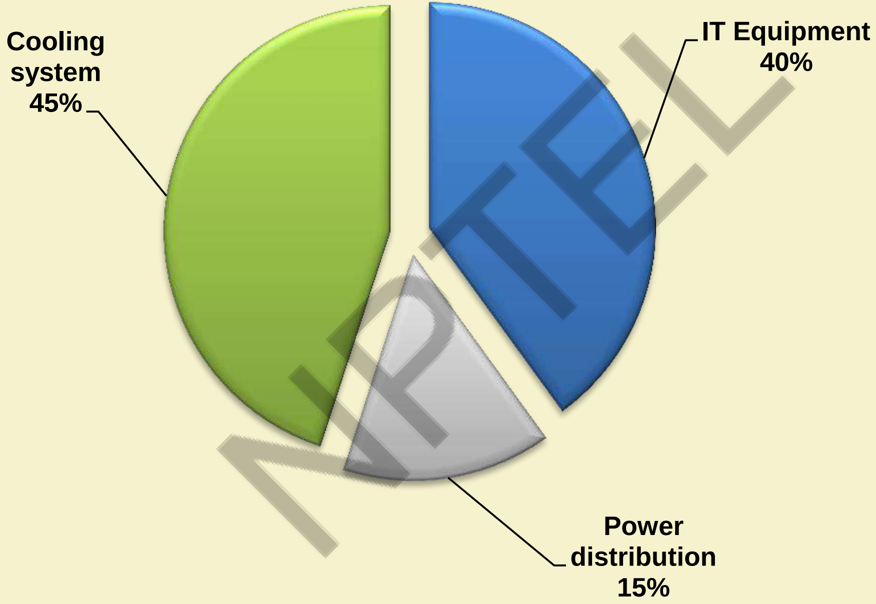
Importance of Energy

- Increased computing demand
  - Data centers are rapidly growing
  - Consume 10 to 100 times more energy per square foot than a typical office building

- Energy cost dynamics
  - Energy accounts for 10% of data center operational expenses (OPEX) and can rise to 50% in the next few years
  - Accompanying cooling system costs $2-$5 million per year

Ref: Dzmitry Kliazovich, University of Luxembourg

Typical Data Center Energy Consumption

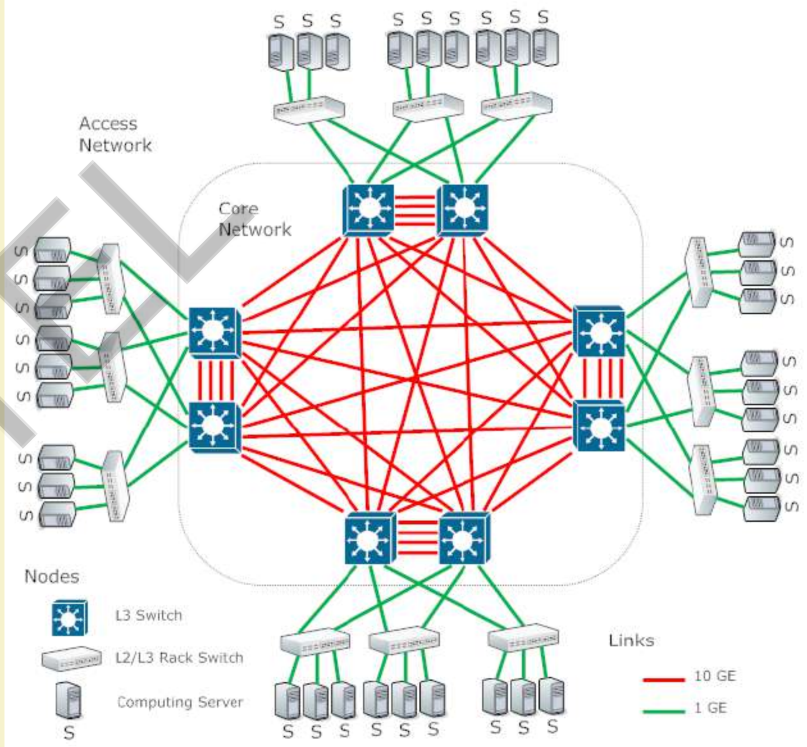


Ref: *Dzmitry Kliazovich, University of Luxembourg*

DC Architecture - Past

Two-tier DC architecture

- Access and Core layers
- 1 GE and 10 GE links
- Full mesh core network
- Load balancing using ICMP

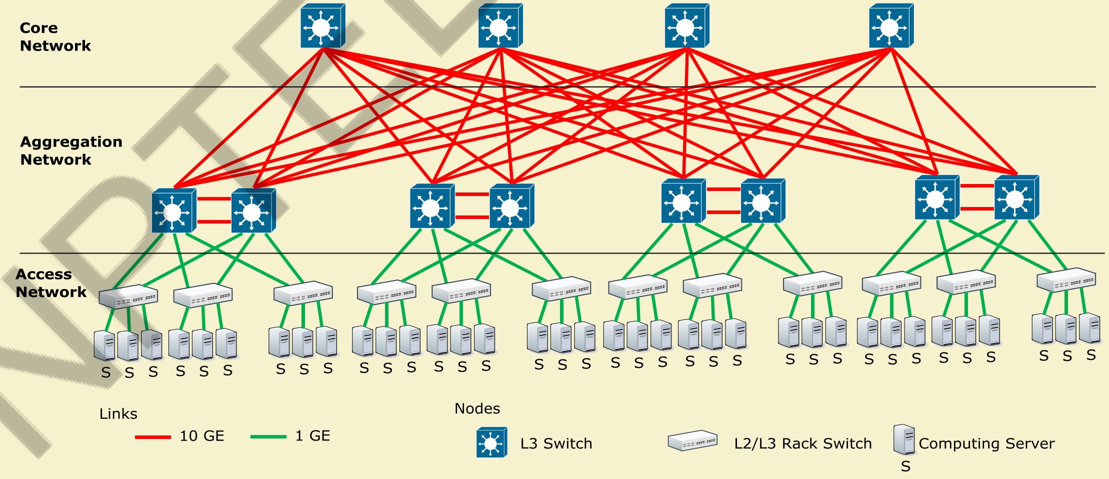


Ref: Dzmitry Kliazovich, University of Luxembourg

DC Architecture - Present

Three-tier DC architecture

- Most Widely Used Nowadays
- Access, Aggregation, and Core layers
- Scales to over 10,000 servers

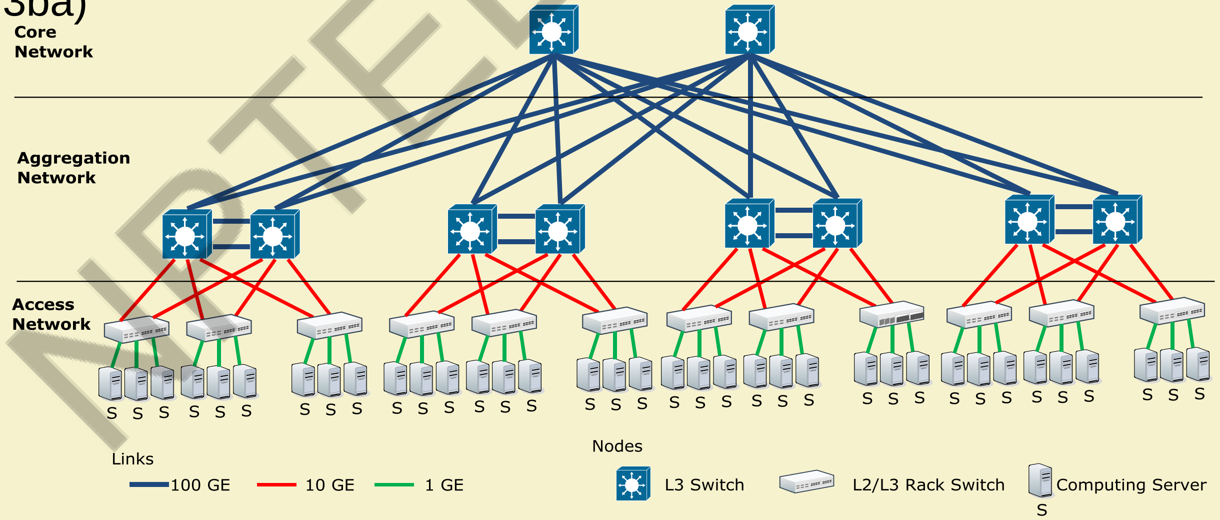


Ref: Dzmitry Kliazovich, University of Luxembourg

DC Architecture - Present

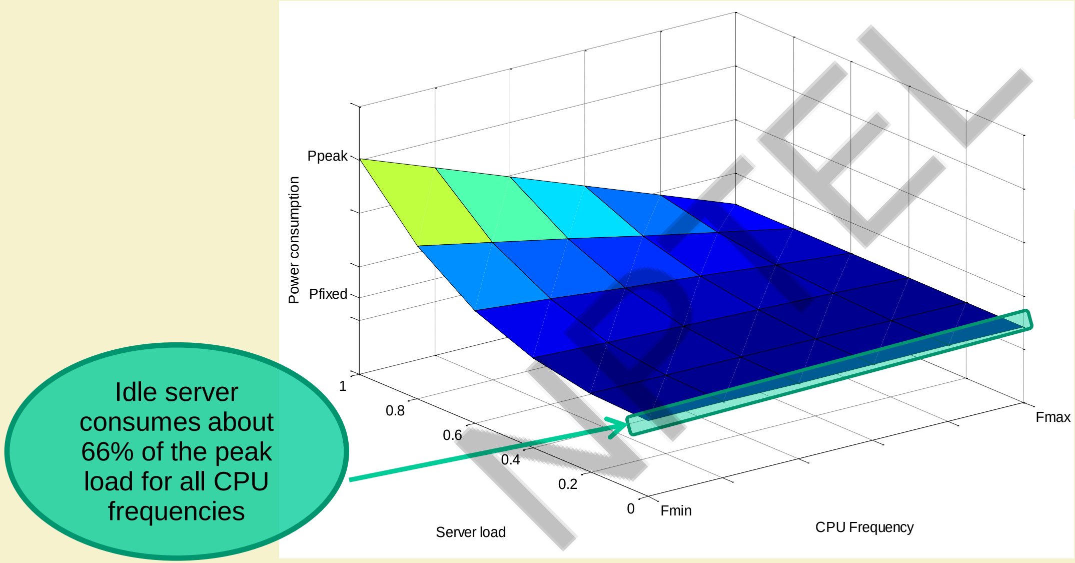
Three-tier High-Speed architecture

- Increased core network bandwidth
- 2-way ECMP load balancing
- 100 GE standard (IEEE 802.3ba)



Ref: Dzmitry Kliazovich, University of Luxembourg

DC Server Energy Model



Idle server consumes about 66% of the peak load for all CPU frequencies

$$P = P_{fixed} + P_f * f^3$$

memory modules, disks, I/O resources

CPU

Ref: Dzmitry Kliazovich, University of Luxembourg

DC Network Switches’ Energy Model

$$P_{switch} = P_{chasis} + n_{linecards} \cdot P_{linecard} + \sum_{i=0}^{R} n_{ports.r} \cdot P_r$$

Chassis
~ 36%

+

Linecards
~ 53%

+

Port transceivers
~ 11%

Ref: Dzmitry Kliazovich, University of Luxembourg

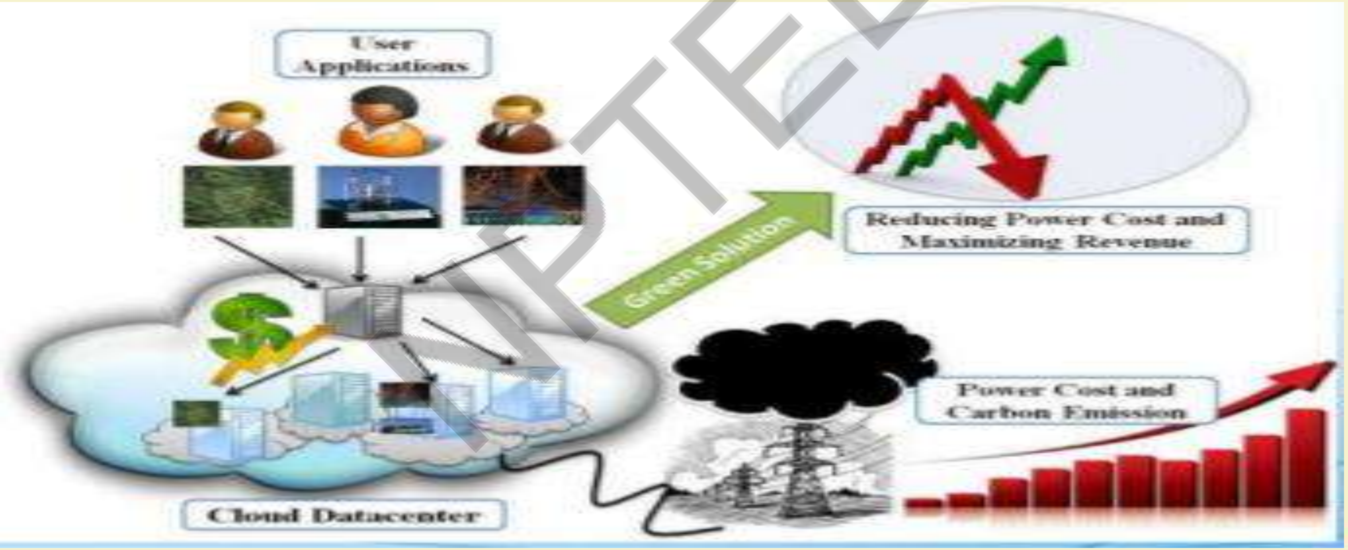
Impact of Cloud DC on Environment

- Data centers are not only expensive to maintain, but also unfriendly to the environment.
- Carbon emission due to Data Centers worldwide is now more than both Argentina and the Netherlands emission.
- High energy costs and huge carbon footprints are incurred due to the massive amount of electricity needed to power and cool the numerous servers hosted in these data centers.

Source: Internet

Performance <-> Energy Efficiency

As energy costs are increasing while availability decreases, there is a need to shift focus from optimizing data center resource management for pure performance alone to optimizing for energy efficiency while maintaining high service level performance.



User Applications
Green Solution
Reducing Power Cost and Maximizing Revenue
Power Cost and Carbon Emission
Cloud Datacenter

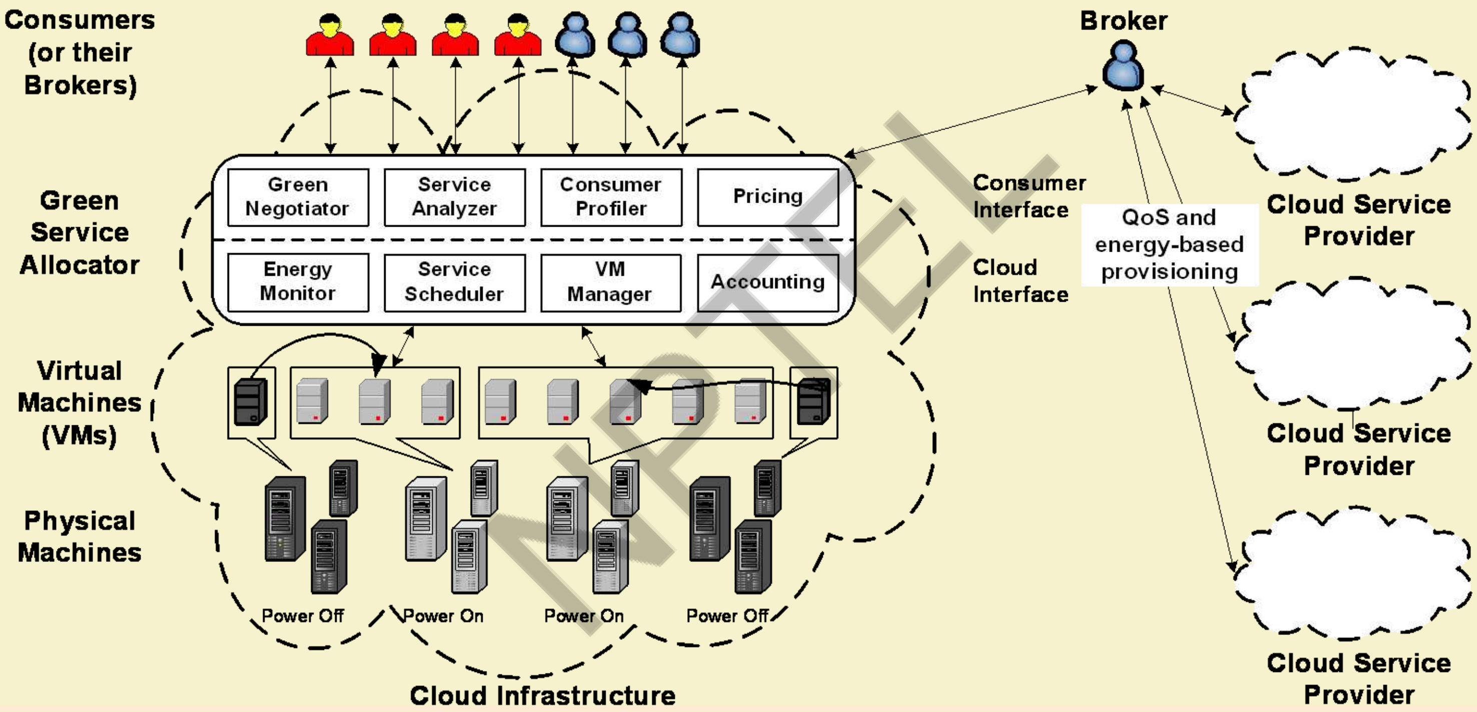
Source: Internet

CSP Initiatives

- Cloud service providers need to adopt measures to ensure that their profit margin is not dramatically reduced due to high energy costs.
- Amazon.com's estimate the energy-related costs of its data centers amount to 42% of the total budget that include both direct power consumption and the cooling infrastructure amortized over a 15-year period.
- Google, Microsoft, and Yahoo are building large data centers in barren desert land surrounding the Columbia River, USA to exploit cheap hydroelectric power.

Source: Internet

A Typical Green Cloud Architecture



Source: Internet

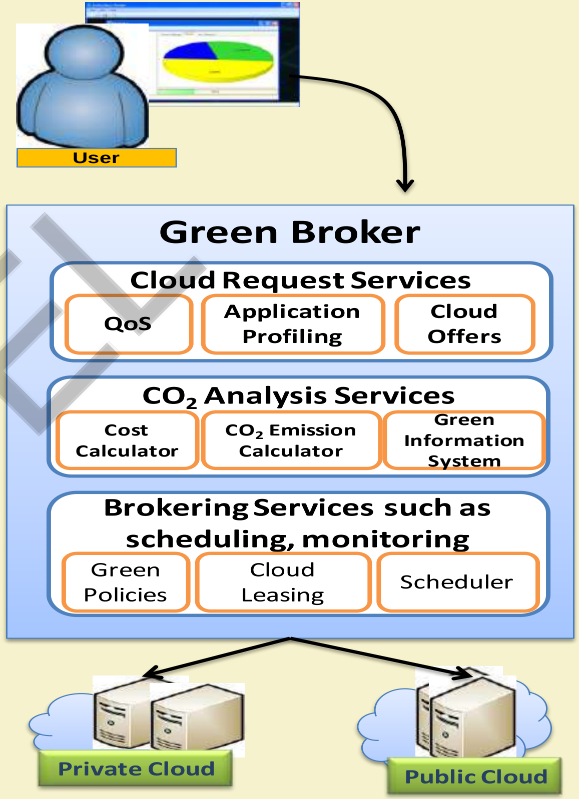
Green Broker

A typical Cloud broker

- Lease Cloud services
- Schedule applications

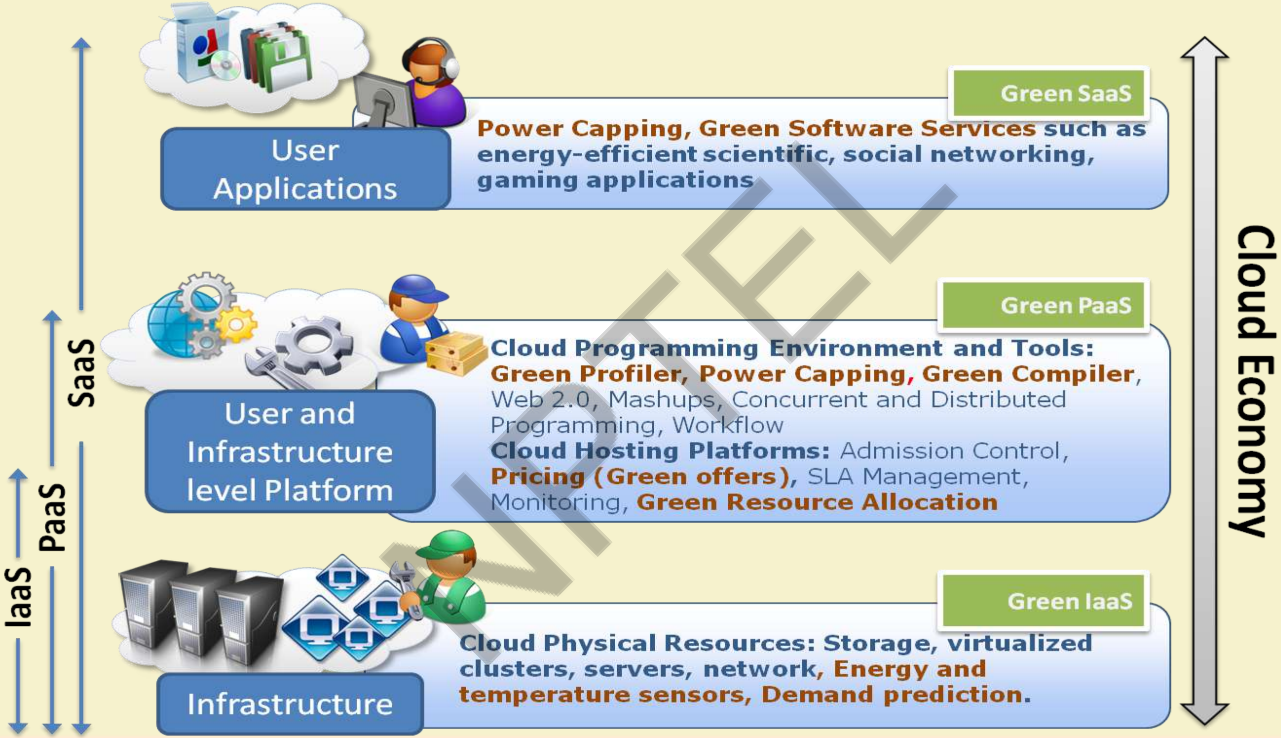
Green Broker

- 1st layer: Analyze user requirements
- 2nd layer: Calculates cost and carbon footprint of services
- 3rd layer: Carbon aware scheduling



Source: Internet

Green Middleware



SaaS

PaaS

IaaS

Cloud Economy

User Applications

Green SaaS

Power Capping, Green Software Services such as energy-efficient scientific, social networking, gaming applications

User and Infrastructure level Platform

Green PaaS

Cloud Programming Environment and Tools: Green Profiler, Power Capping, Green Compiler, Web 2.0, Mashups, Concurrent and Distributed Programming, Workflow

Cloud Hosting Platforms: Admission Control, Pricing (Green offers), SLA Management, Monitoring, Green Resource Allocation

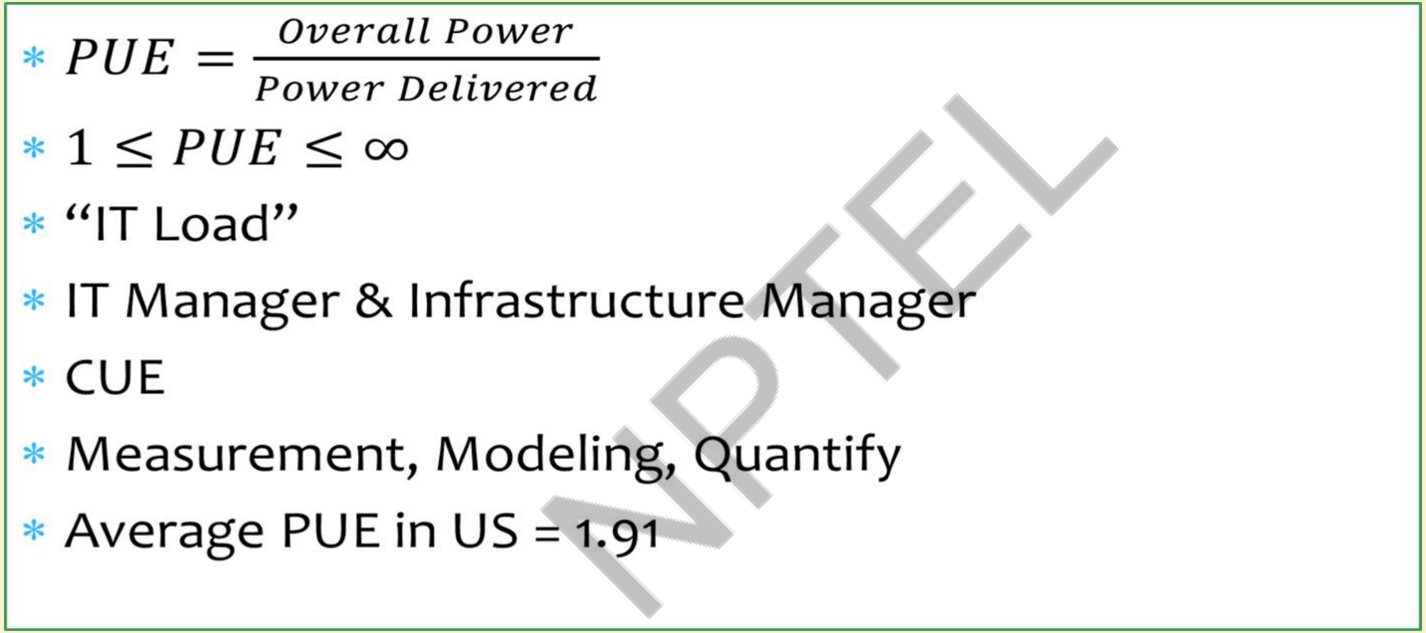
Infrastructure

Green IaaS

Cloud Physical Resources: Storage, virtualized clusters, servers, network, Energy and temperature sensors, Demand prediction.

Source: Internet

Power Usage Effectiveness (PUE)



- $PUE = \frac{Overall\ Power}{Power\ Delivered}$
- $1 \leq PUE \leq \infty$
- “IT Load”
- IT Manager & Infrastructure Manager
- CUE
- Measurement, Modeling, Quantify
- Average PUE in US = 1.91

Source: Internet

Conclusions

- Clouds are essentially Data Centers hosting application services offered on a subscription basis. However, they consume high energy to maintain their operations.

  => high operational cost + environmental impact
- Presented a Carbon Aware Green Cloud Framework to improve the carbon footprint of Cloud computing.
- Open Issues: Lots of research to be carried out for Maximizing Efficiency of Green Data Centers and Developing Regions to benefit the most.

Source: Internet

Thank You!

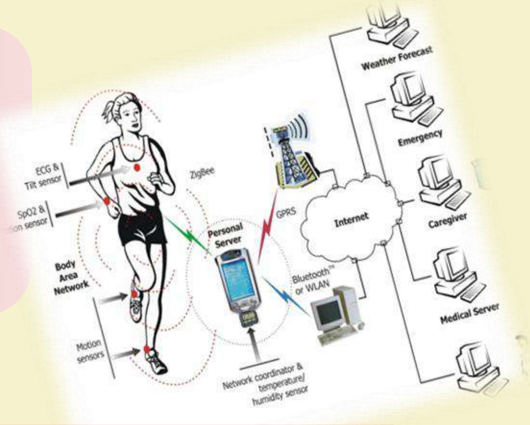
CLOUD COMPUTING

Sensor Cloud Computing

Prof. Soumya K Ghosh
Department of Computer Science and Engineering
IIT KHARAGPUR

Motivation

- *Increasing adoption of sensing technologies (e.g., RFID, cameras, mobile phones)*
- *Internet has become a source of real time information (e.g., through blogs, social networks, live forums) for events happening around us*

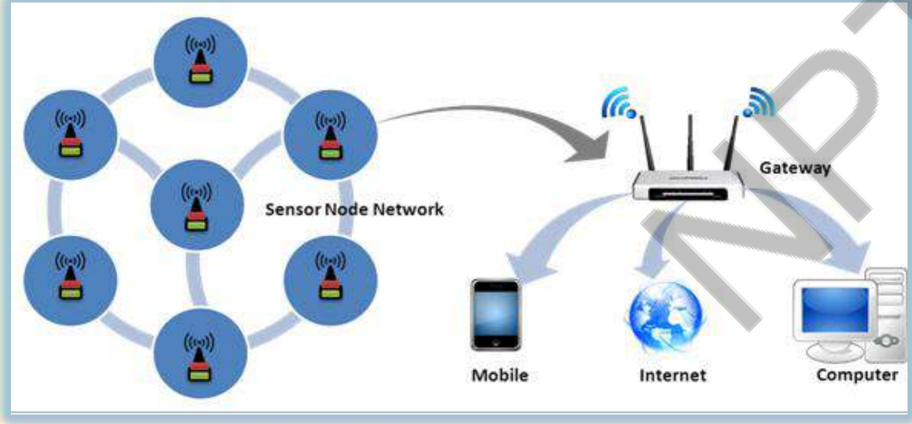


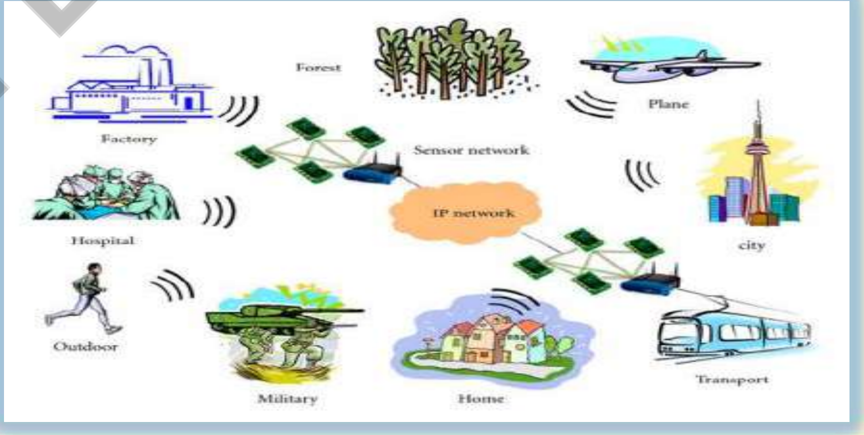
- *Cloud computing has emerged as an attractive solution for dealing with the "Big Data" revolution*
- *By combining data obtained from sensors with that from the internet, we can potentially create a demand for resources that can be appropriately met by the cloud*

Ref: Tan, Kian-Lee. "What's next?: Sensor+ cloud!?." Proceedings of the Seventh International Workshop on Data Management for Sensor Networks. ACM, 2010

Wireless Sensor Network (WSNs)

- Seamlessly couples the physical environment with the digital world
- Sensor nodes are small, low power, low cost, and provide multiple functionalities
  - Sensing capability, processing power, memory, communication bandwidth, battery power.
- In aggregate, sensor nodes have substantial data acquisition and processing capability
- Useful in many application domains – Environment, Healthcare, Education, Defense, Manufacturing, Smart Home, etc.





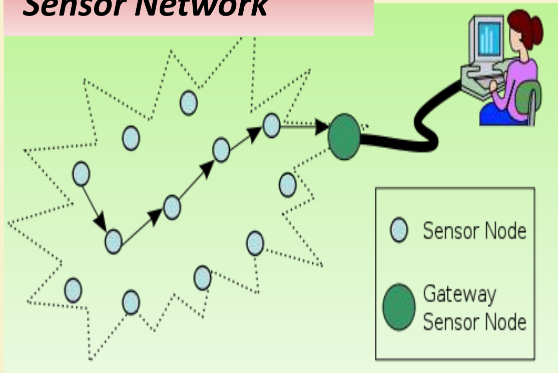
Limitations of Sensor Networks

- Very challenging to scale sensor networks to large sizes
- Proprietary vendor-specific designs. Difficult for different sensor networks to be interconnected
- Sensor data cannot be easily shared by different groups of users.
- Insufficient computational and storage resources to handle large-scale applications.
- Used for fixed and specific applications that cannot be easily changed once deployed.
- Slow adoption of large-scale sensor network applications.

Limitations of Cloud Computing!

- The immense power of the Cloud can only be fully exploited if it is seamlessly integrated into our physical lives.
- That means – providing the *real world's* information to the Cloud in *real time* and getting the Cloud to *act and serve us instantly*.
- That is – adding the sensing capability to the Cloud

Sensor Network



What is missing?

Computing Platform

Cloud Server

Mobile Computing

Applications

Cloud Security

Cloud Storage

Cloud Economics

Social Networks

Codes

Services

A Motivating Scenario!

1. Lets go to the *mountain peak*!

2. Sounds Good!

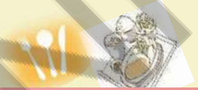
I. Please take your lunch as you appear hungry!

II. Carry drinking water – Water at that region is contaminated

III. Use anti-UV skin cream

3. Map to nearest food outlets

4. Take pictures of restaurants and send images



5. Menus of restaurants and recommended foods!

6. Your friend is at nearby restaurant.. Go catch up with her!

Source: Internet

Few insight from the example!

- Cell phone records the tourist's gestures and activates applications such as camera, microphone, etc.
- Cell phone produces very swift responses in real time after:
  - Processing geographical data
  - Acquiring tourist's physiological data from wearable physiological
  - Sensors (blood sugar, precipitation, etc.) and cross-comparing it with his medical records
  - Speech recognition
  - Image processing of restaurant's logos and accessing their internet-based profiles
  - Accessing tourist's social network profiles to find out his friends

Fact : the cell phone cannot perform so much tasks !

Ref: http://www.ntu.edu.sg/intellisys

Need to integrate Sensors with Cloud!

- Acquisition of data feeds from numerous body area (blood sugar, heat, perspiration, etc) and wide area (water quality, weather monitoring, etc.) sensor networks in real time.
- Real-time processing of heterogeneous data sources in order to make critical decisions.
- Automatic formation of workflows and invocation of services on the cloud one after another to carry out complex tasks.
- Highly swift data processing using the immense processing power of the cloud to provide quick response to the user.

What is Sensor Cloud Computing?

An infrastructure that allows truly pervasive computation using sensors as interface between physical and cyber worlds, the data-compute clusters as the cyber backbone and the internet as the communication medium

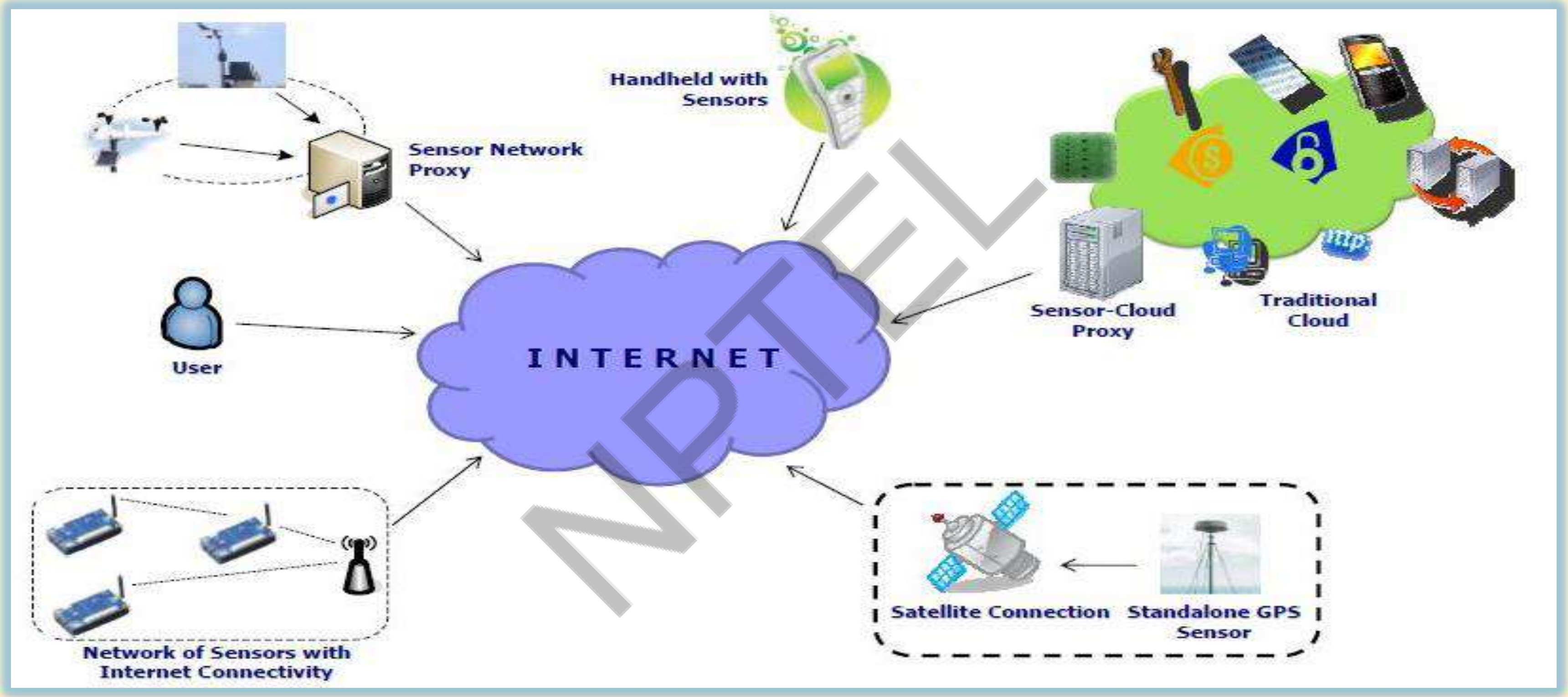
- It integrates large-scale sensor networks with sensing applications and cloud computing infrastructures.
- It collects and processes data from various sensor networks.
- Enables large-scale data sharing and collaborations among users and applications on the cloud.
- Delivers cloud services via sensor-rich devices.
- Allows cross-disciplinary applications that span organizational boundaries.

Sensor Cloud?

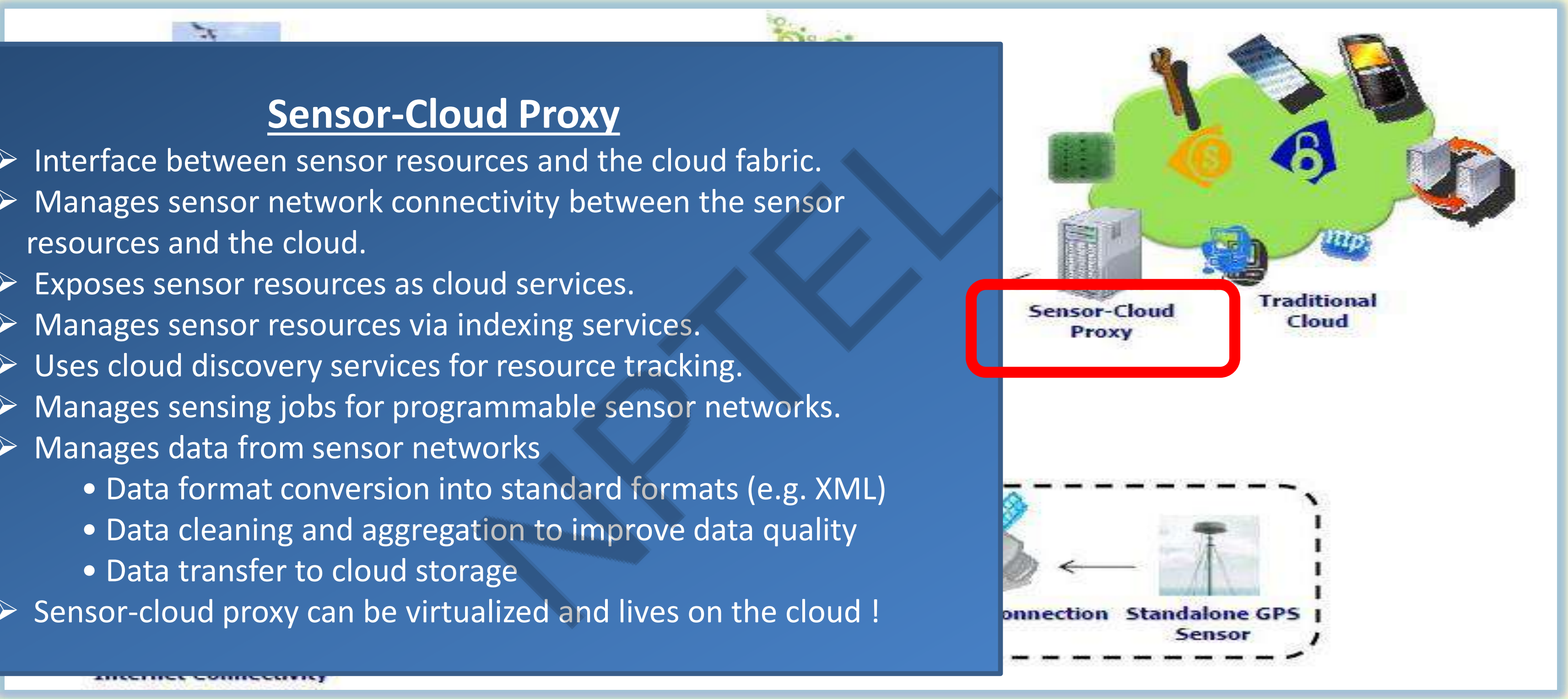
- Enables users to easily collect, access, process, visualize, archive, share and search large amounts of sensor data from different applications.
- Supports complete sensor data life cycle from data collection to the bac
- Va using cor
- Allows sharing of sensor resources by different users and applications under flexible usage scenarios.
- Enables sensor devices to handle specialized processing tasks.

Sensor cloud enables different networks, spread in a huge geographical area, to connect together and be employed simultaneously by multiple users on demand

Overview of Sensor-Cloud Framework



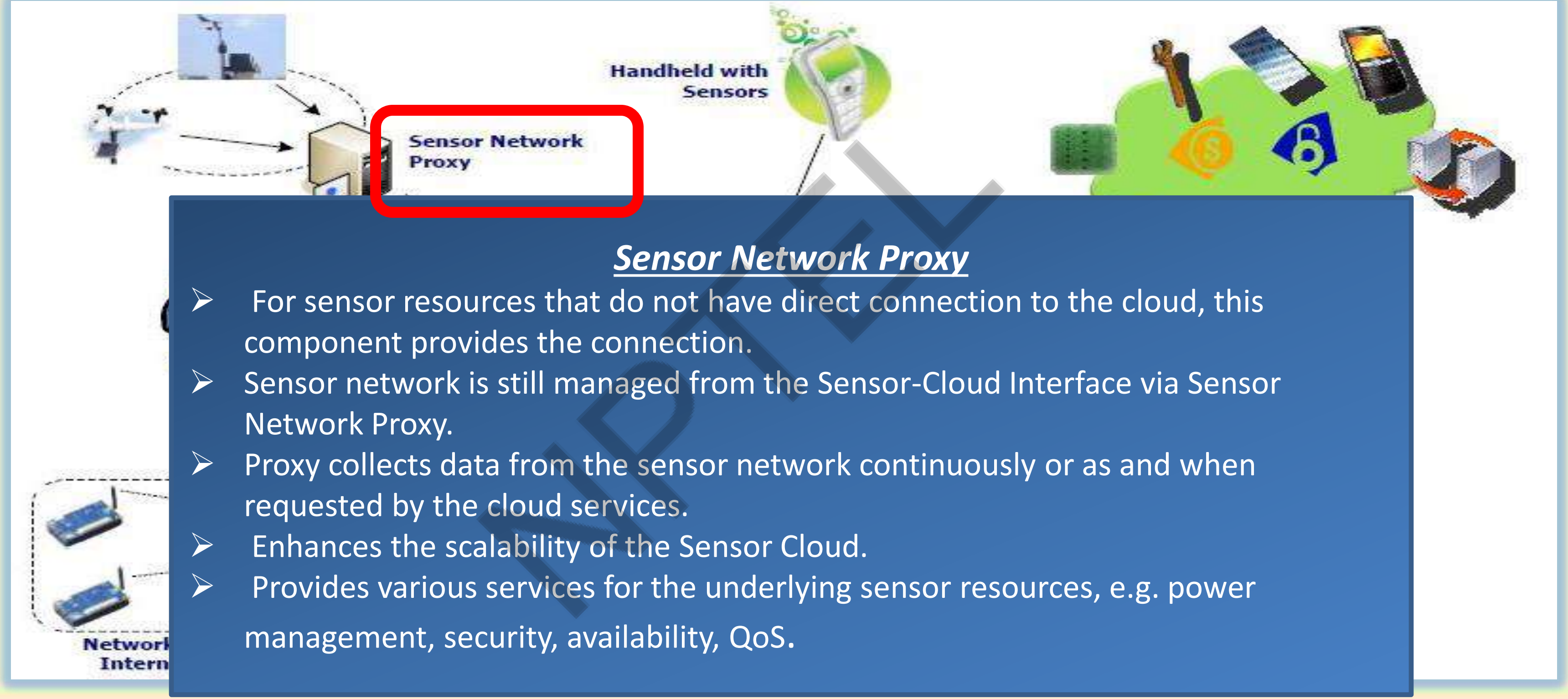
Overview of Sensor-Cloud Framework



Sensor-Cloud Proxy

- Interface between sensor resources and the cloud fabric.
- Manages sensor network connectivity between the sensor resources and the cloud.
- Exposes sensor resources as cloud services.
- Manages sensor resources via indexing services.
- Uses cloud discovery services for resource tracking.
- Manages sensing jobs for programmable sensor networks.
- Manages data from sensor networks
  - Data format conversion into standard formats (e.g. XML)
  - Data cleaning and aggregation to improve data quality
  - Data transfer to cloud storage
- Sensor-cloud proxy can be virtualized and lives on the cloud !

Overview of Sensor-Cloud Framework



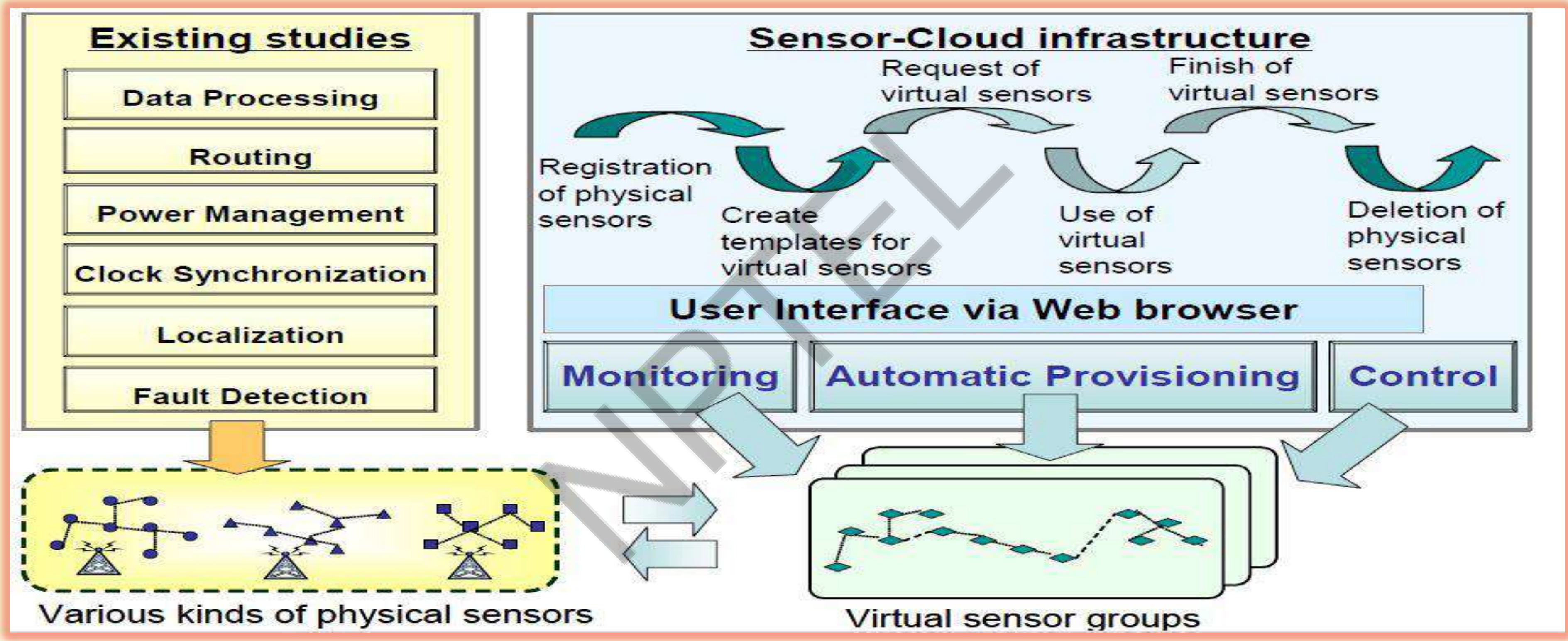
Sensor Network Proxy

- For sensor resources that do not have direct connection to the cloud, this component provides the connection.
- Sensor network is still managed from the Sensor-Cloud Interface via Sensor Network Proxy.
- Proxy collects data from the sensor network continuously or as and when requested by the cloud services.
- Enhances the scalability of the Sensor Cloud.
- Provides various services for the underlying sensor resources, e.g. power management, security, availability, QoS.

Another Use case...

- Traffic flow sensors are widely deployed in large numbers in places/ cities.
- These sensors are mounted on traffic lights and provide real-time traffic flow data.
- Drivers can use this data to better plan their trips.
- In addition, if the traffic flow sensors are augmented with low-cost humidity and temperature sensors, they can provide a customized and local view of temperature and heat index data on demand.
- The national weather service, on the other hand, uses a single weather station to collect environmental data for a large area, which might not accurately represent an entire region.

Overview of Sensor Cloud Infrastructure

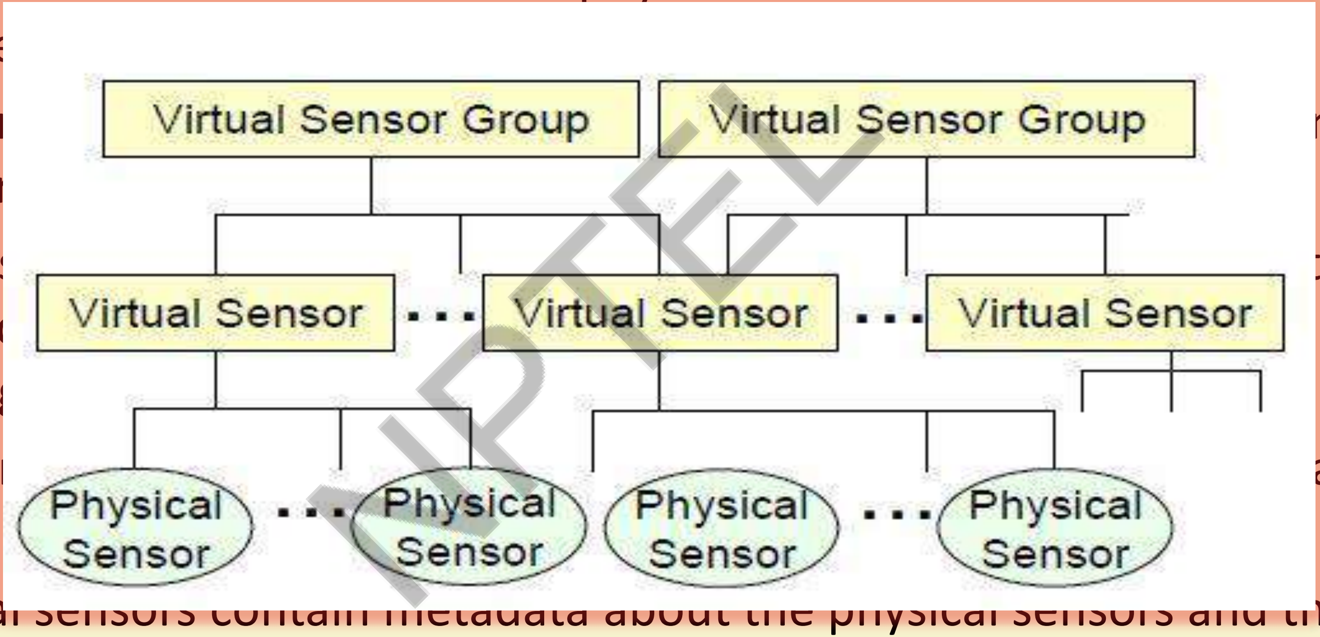


Existing studies
Data Processing
Routing
Power Management
Clock Synchronization
Localization
Fault Detection
Various kinds of physical sensors
Sensor-Cloud infrastructure
Registration of physical sensors
Create templates for virtual sensors
Request of virtual sensors
Use of virtual sensors
Finish of virtual sensors
Deletion of physical sensors
User Interface via Web browser
Monitoring
Automatic Provisioning
Control
Virtual sensor groups

Madoka et al. *"Sensor-Cloud Infrastructure Physical Sensor Management with Virtualized Sensors on Cloud Computing "*

Virtual Sensors?

- A virtual sensor is an emulation of a physical sensor that obtains its data from unde
- Virtual se nd location t
- In wireles a time and computin
- To overco are of the co
- The virtual sensors contain metadata about the physical sensors and the user currently holding that virtual sensor.



Virtual Sensor Group

Virtual Sensor Group

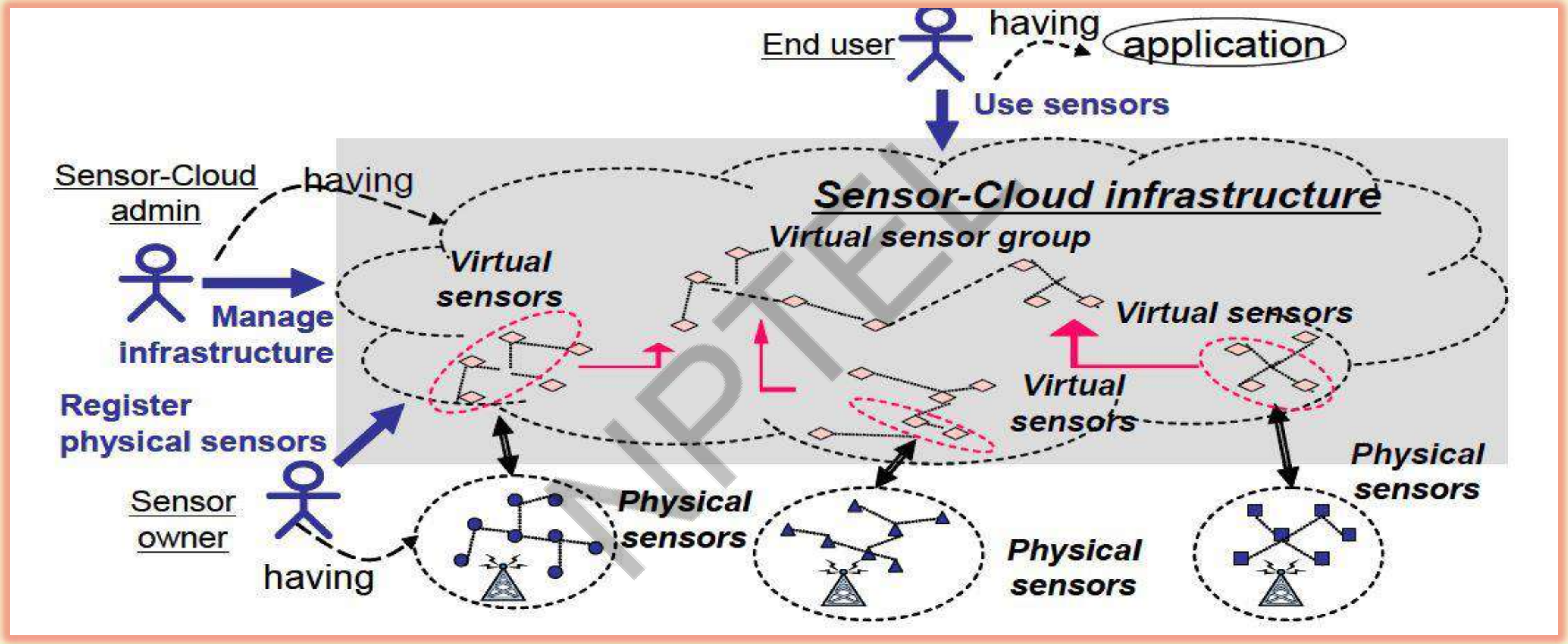
Virtual Sensor ... Virtual Sensor ... Virtual Sensor

Physical Sensor ... Physical Sensor

Physical Sensor ... Physical Sensor

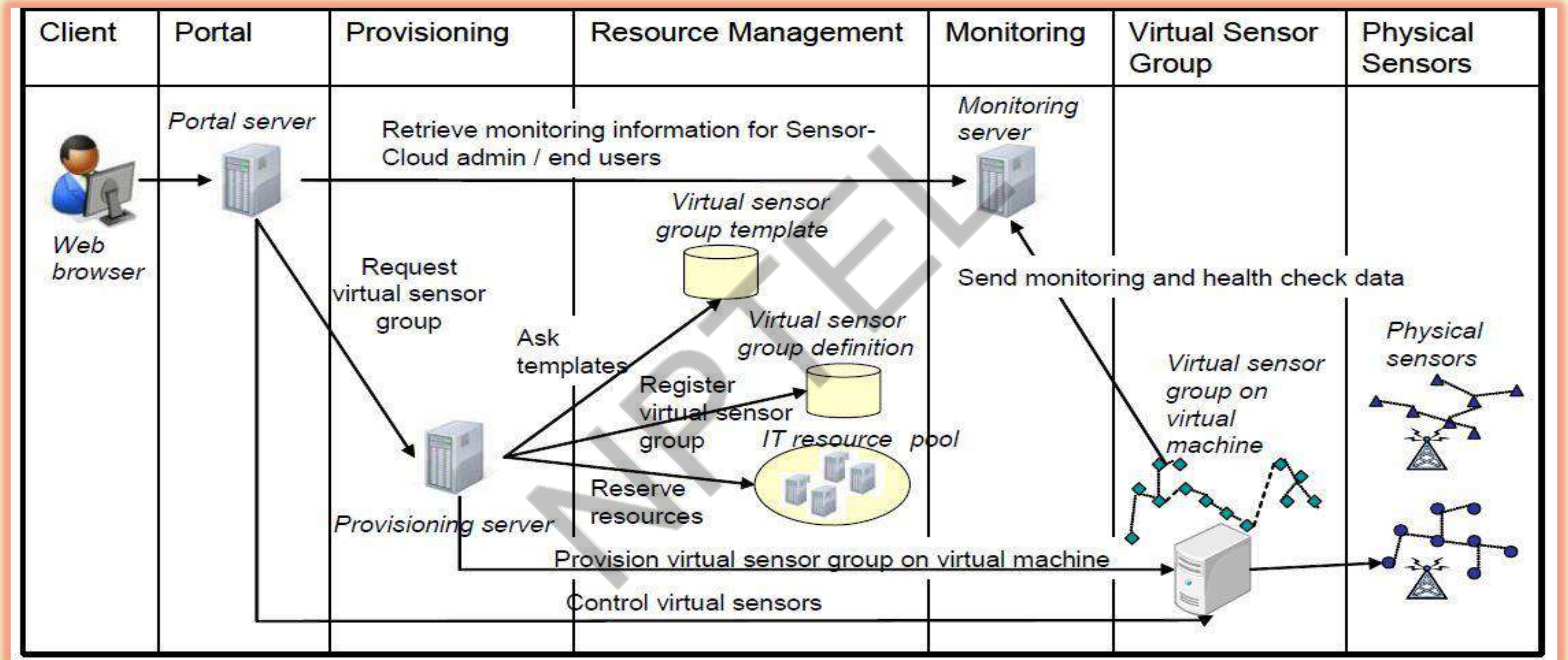
***Madoka et al. "Sensor-Cloud Infrastructure** Physical Sensor Management with Virtualized Sensors on Cloud Computing "*

Relationship among Actors and Sensor Cloud Infrastructure



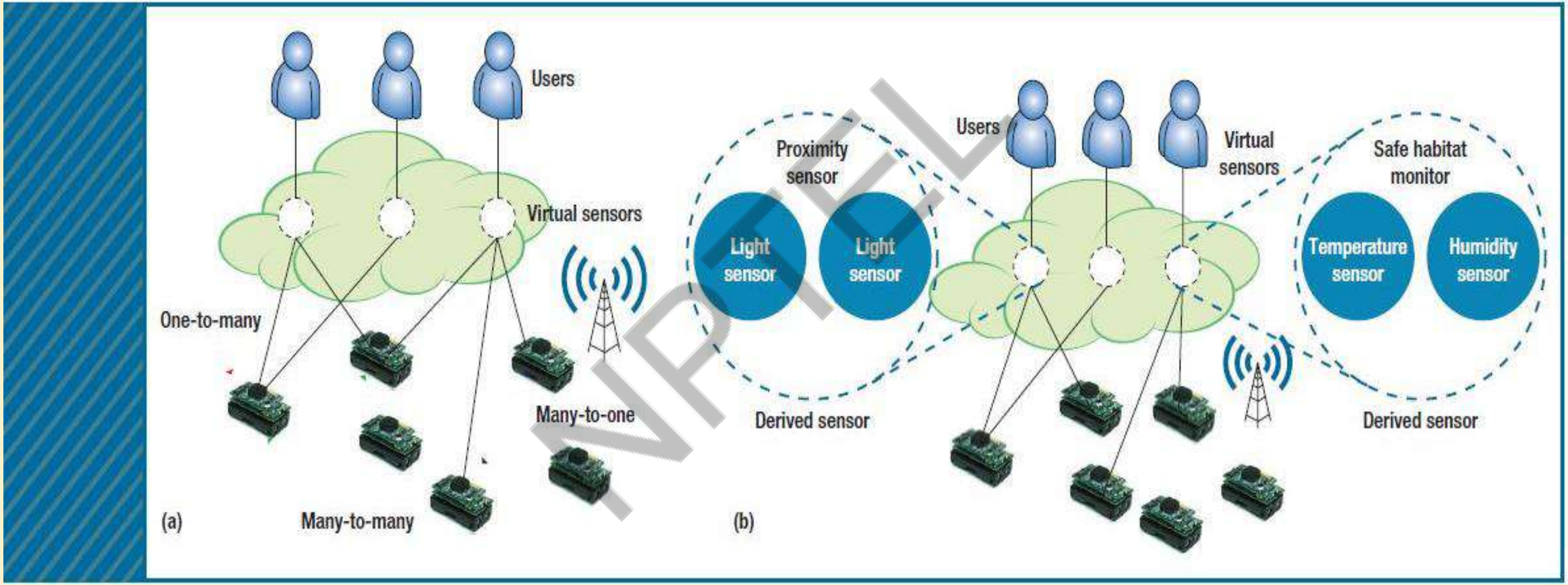
***Madoka et al. "Sensor-Cloud Infrastructure** Physical Sensor Management with Virtualized Sensors on Cloud Computing "*

System Architecture of Sensor Cloud Infrastructure


Client
Portal
Provisioning
Resource Management
Monitoring
Virtual Sensor Group
Physical Sensors
Web browser
Portal server
Retrieve monitoring information for Sensor-Cloud admin / end users
Monitoring server
Virtual sensor group template
Send monitoring and health check data
Request virtual sensor group
Ask templates
Virtual sensor group definition
Register virtual sensor group
IT resource pool
Physical sensors
Virtual sensor group on virtual machine
Reserve resources
Provisioning server
Provision virtual sensor group on virtual machine
Control virtual sensors

Virtual Sensor Configurations

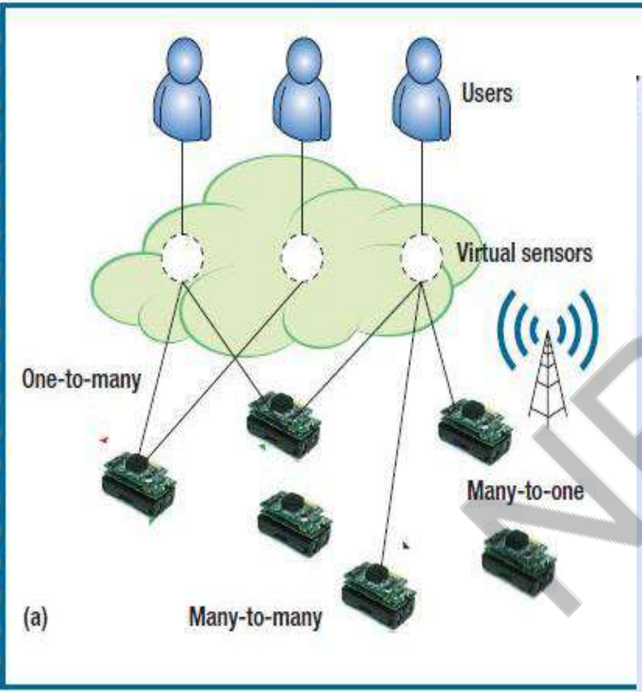
(a) one-to-many, many-to-one, and many-to-many, and (b) derived



Sanjay et al. "Sensor Cloud: A Cloud of Virtual Sensors" , IEEE Software, 2014

Virtual Sensor Configurations

(a) one-to-many, many-to-one, and many-to-many, and (b) derived

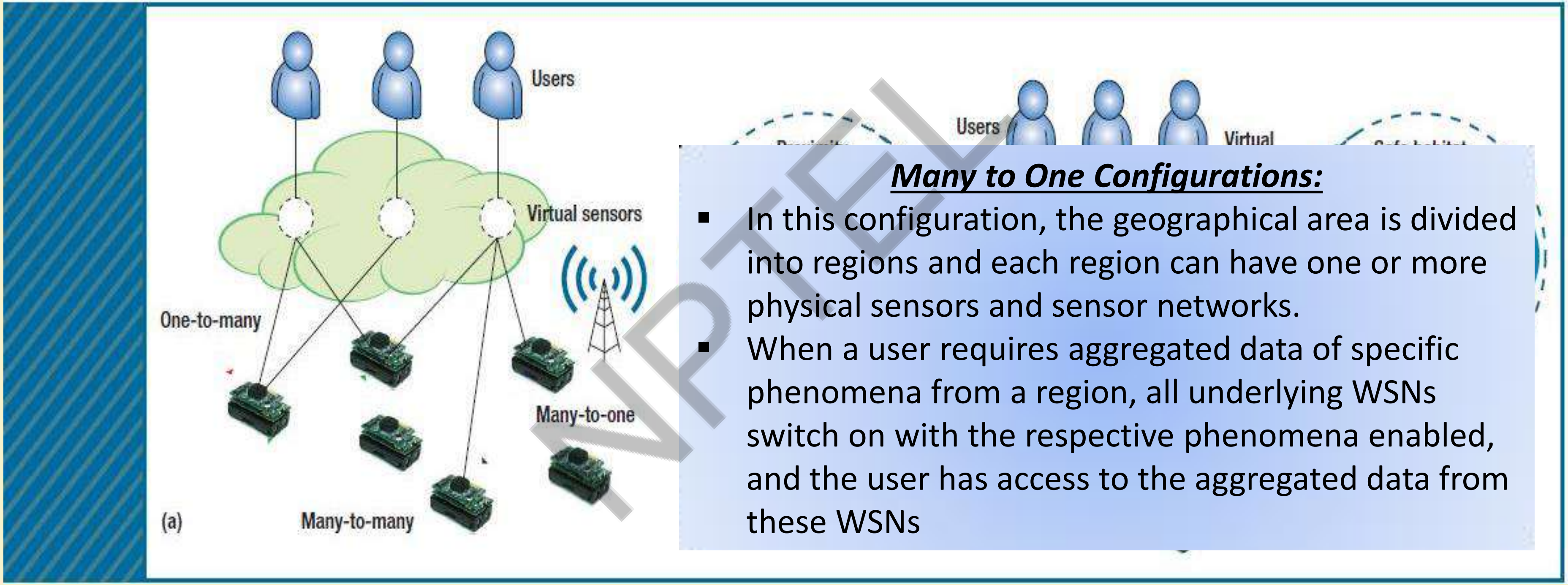


One to Many Configurations:

- In this configuration, one physical sensor corresponds to many virtual sensors.
- Although individual users own the virtual image, the underlying physical sensor is shared among all the virtual sensors accessing it.
- The middleware computes the physical sensor's sampling duration and frequency by taking into account all the users; it re-evaluates the duration and frequency when new users join or existing users leave the system.

Virtual Sensor Configurations

(a) one-to-many, many-to-one, and many-to-many, and (b) derived

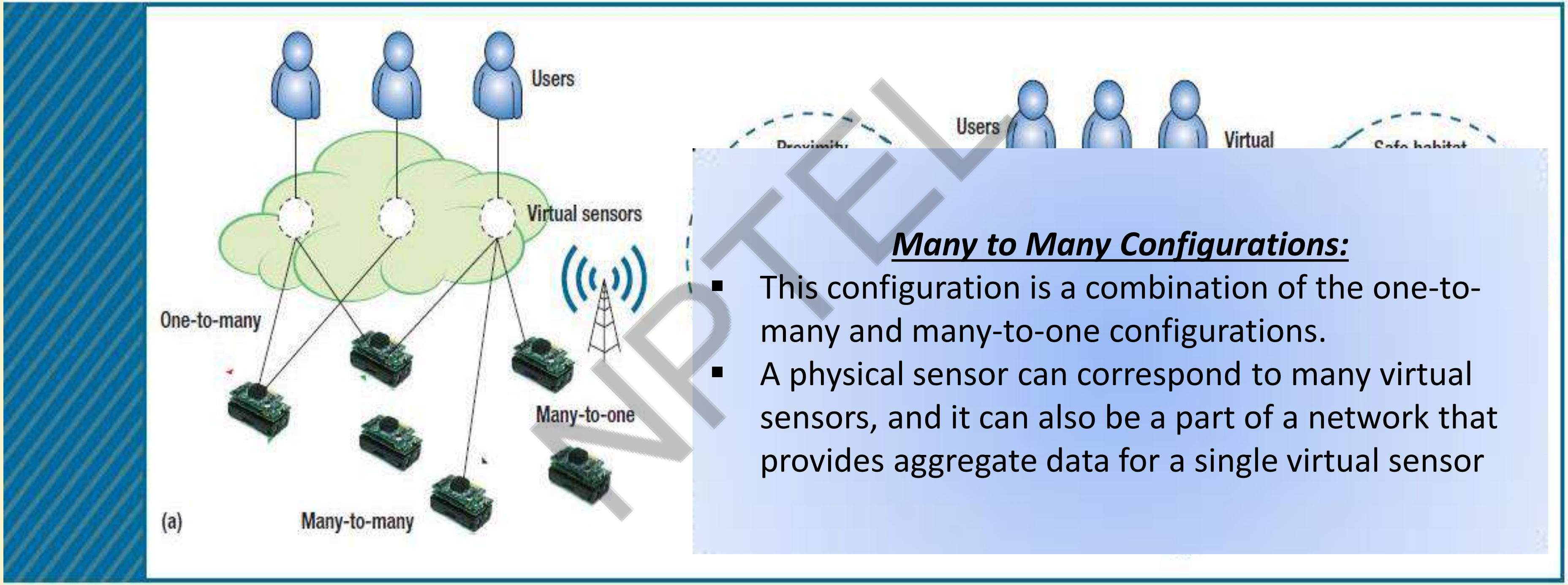


Many to One Configurations:

- In this configuration, the geographical area is divided into regions and each region can have one or more physical sensors and sensor networks.
- When a user requires aggregated data of specific phenomena from a region, all underlying WSNs switch on with the respective phenomena enabled, and the user has access to the aggregated data from these WSNs

Virtual Sensor Configurations

(a) one-to-many, many-to-one, and many-to-many, and (b) derived



Many to Many Configurations:

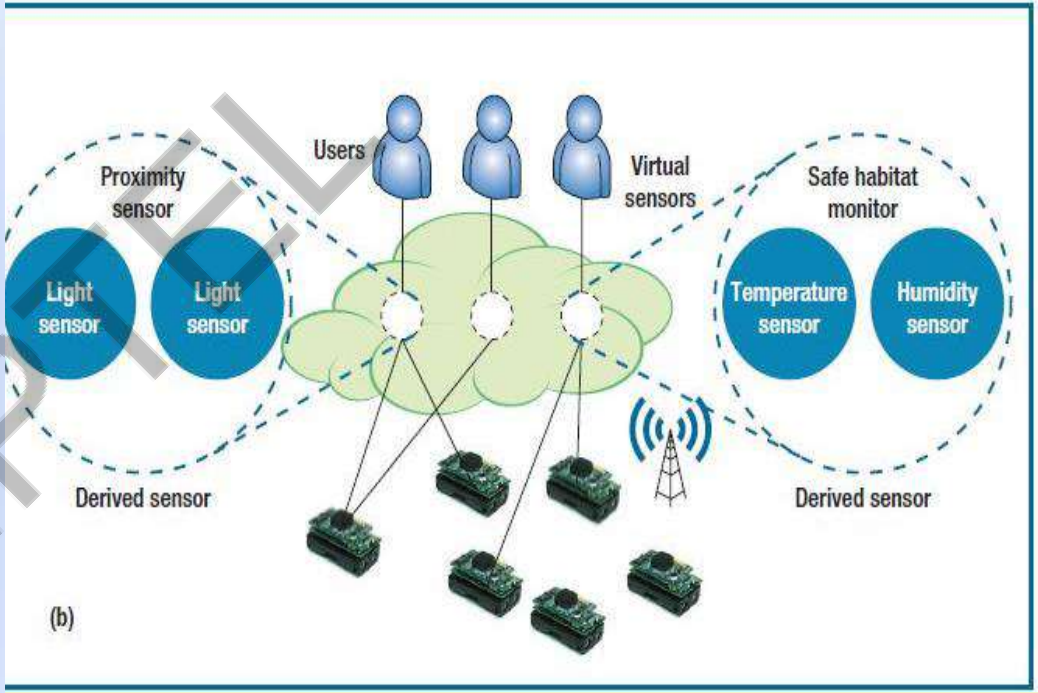
- This configuration is a combination of the one-to-many and many-to-one configurations.
- A physical sensor can correspond to many virtual sensors, and it can also be a part of a network that provides aggregate data for a single virtual sensor

Virtual Sensor Configurations

(a) one-to-many, many-to-one, and many-to-many, and (b) derived

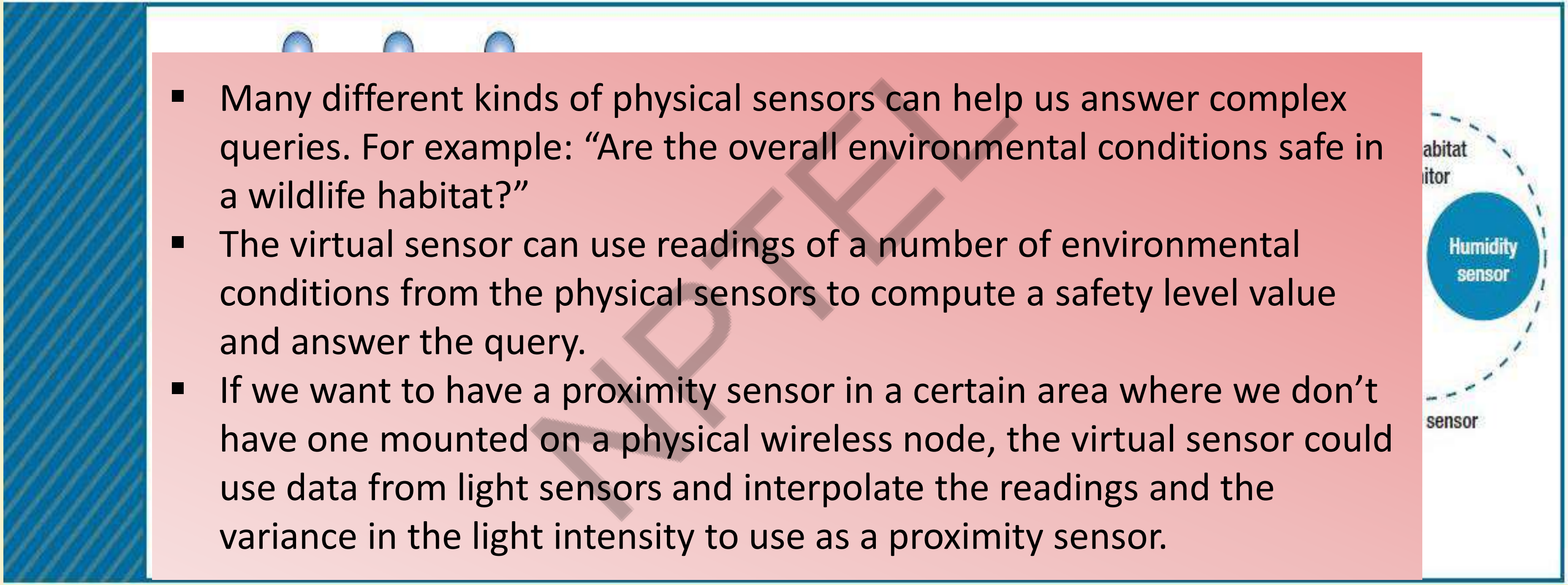
Derived:

- A derived configuration refers to a versatile configuration of virtual sensors derived from a combination of multiple physical sensors.
- This configuration can be seen as a generalization of the other three configurations, though, the difference lies in the types of physical sensors with which a virtual sensor communicates.
- While in the derived configuration, the virtual sensor communicates with multiple sensor types; in the other three configurations, the virtual sensor communicates with the same type of physical sensors.
- Derived sensors can be used in two ways: first, to virtually sense complex phenomenon and second, to substitute for sensors that aren't physically deployed.



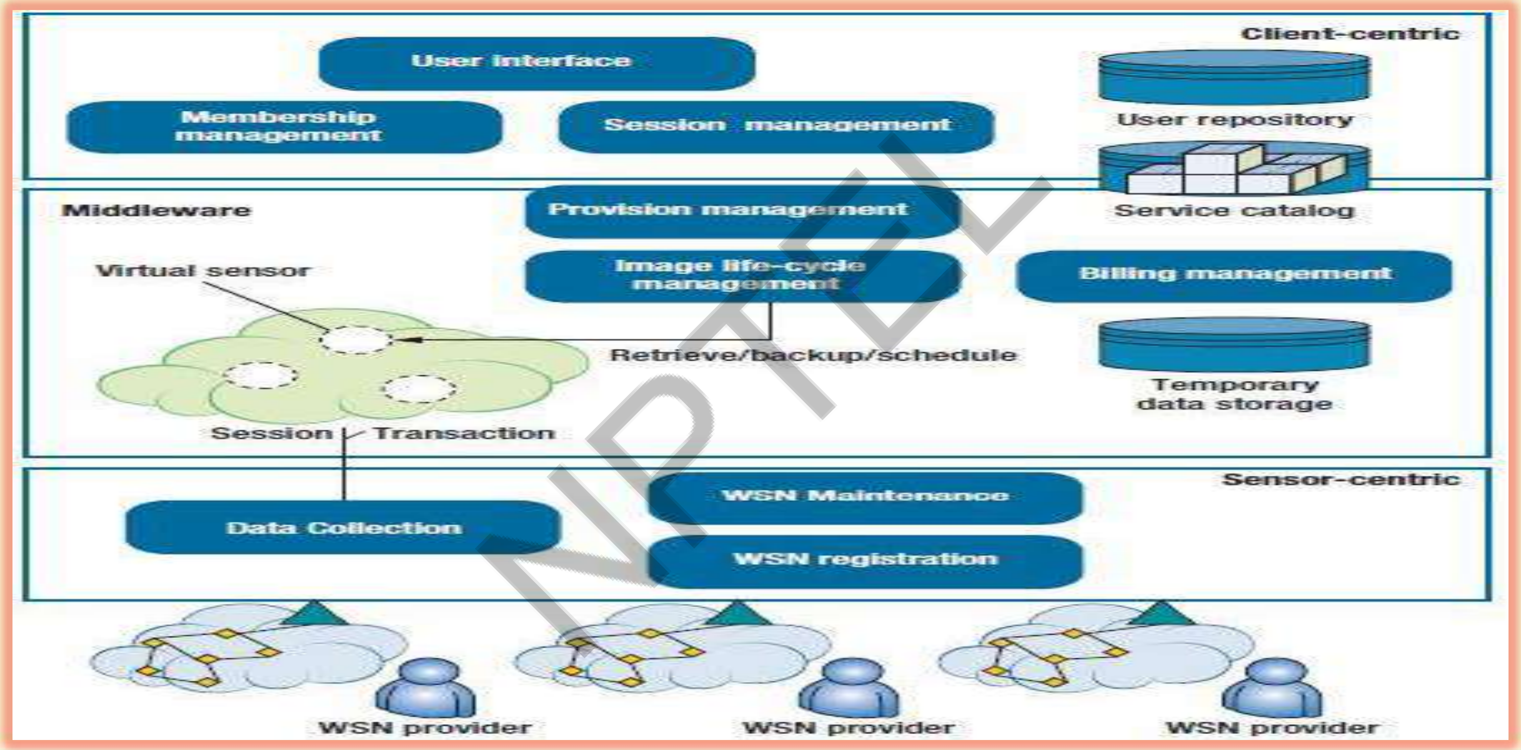
Virtual Sensor Configurations

(a) one-to-many, many-to-one, and many-to-many, and (b) derived



- Many different kinds of physical sensors can help us answer complex queries. For example: “Are the overall environmental conditions safe in a wildlife habitat?”
- The virtual sensor can use readings of a number of environmental conditions from the physical sensors to compute a safety level value and answer the query.
- If we want to have a proximity sensor in a certain area where we don’t have one mounted on a physical wireless node, the virtual sensor could use data from light sensors and interpolate the readings and the variance in the light intensity to use as a proximity sensor.

A Layered Sensor Cloud Architecture



Summary

- Sensor-Cloud infrastructure virtualizes sensors and provides the management mechanism for virtualized sensors
- Sensor-Cloud infrastructure enables end users to create virtual sensor groups dynamically by selecting the templates of virtual sensors or virtual sensor groups with IT resources.
- Sensor-Cloud infrastructure focuses on Sensor system management and Sensor data management
- Sensor clouds aim to take the burden of deploying and managing the network away from the user by acting as a mediator between the user and the sensor networks and providing sensing as a service.

References

- Beng, Lim Hock. "Sensor cloud: Towards sensor-enabled cloud services." *Intelligent Systems Center Nanyang Technological University* (2009)
- ***http://www.ntu.edu.sg/intellisys***
- *Sanjay et al. “Sensor Cloud: A Cloud of Virtual Sensors” , IEEE Software, 2014*
- ***Madoka et al. “Sensor-Cloud Infrastructure*** *Physical Sensor Management with Virtualized Sensors on Cloud Computing”*

Thank You!

IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

NPTEL

CLOUD COMPUTING

IoT Cloud

Prof. Soumya K Ghosh
Department of Computer Science and Engineering
IIT KHARAGPUR

Motivation

- *Increasing adoption of sensing technologies (e.g., RFID, cameras, mobile phones)*
- *Sensor devices are becoming widely available*

Wireless sensor technology play a pivotal role in bridging the gap between the physical and virtual worlds, and enabling things to respond to changes in their physical environment. Sensors collect data from their environment, generating information and raising awareness about context.

Example: *Sensors in an electronic jacket can collect information about changes in external temperature and the parameters of the jacket can be adjusted accordingly*

Truong, Hong-Linh, and Schahram Dustdar. Principles for engineering IoT cloud systems." *IEEE Cloud Computing* 2.2 (2015): 68-76

Internet of Things!

- Extending the current Internet and providing connection, communication, and inter-networking between devices and physical objects, or "Things," is a growing trend that is often referred to as the *Internet of Things*.

- The I th unique ident to-human or hu

rm animal with a biochip transponder, an automobile that has built-in sensors to alert the driver when tire pressure is low -- or any other natural or man-made object that can be assigned an IP address and provided with the ability to transfer data over a network

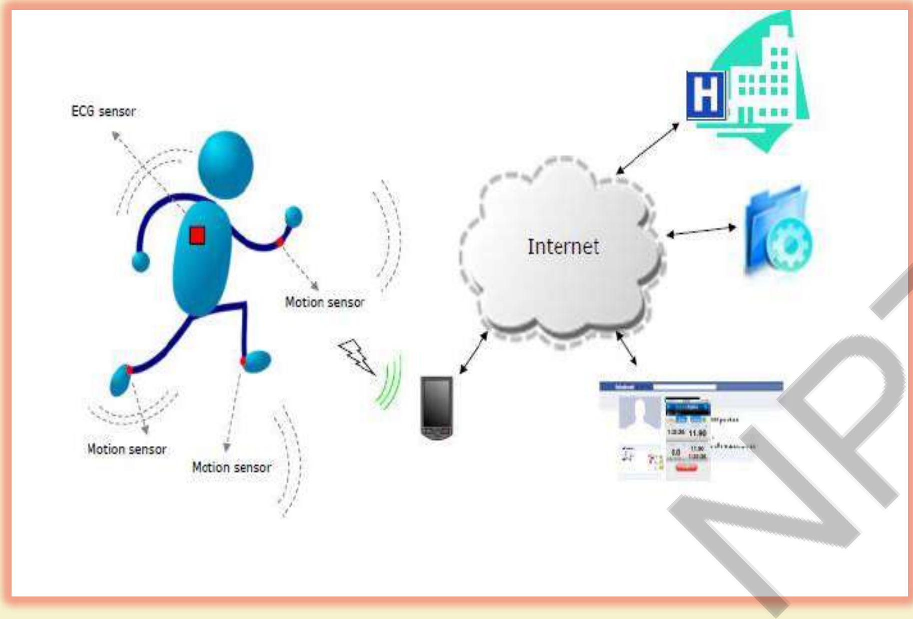
> “The technologies and solutions that enable integration of real world data and services into the current information networking technologies are often described under the umbrella term of the Internet of Things (IoT)”

Source: Internet

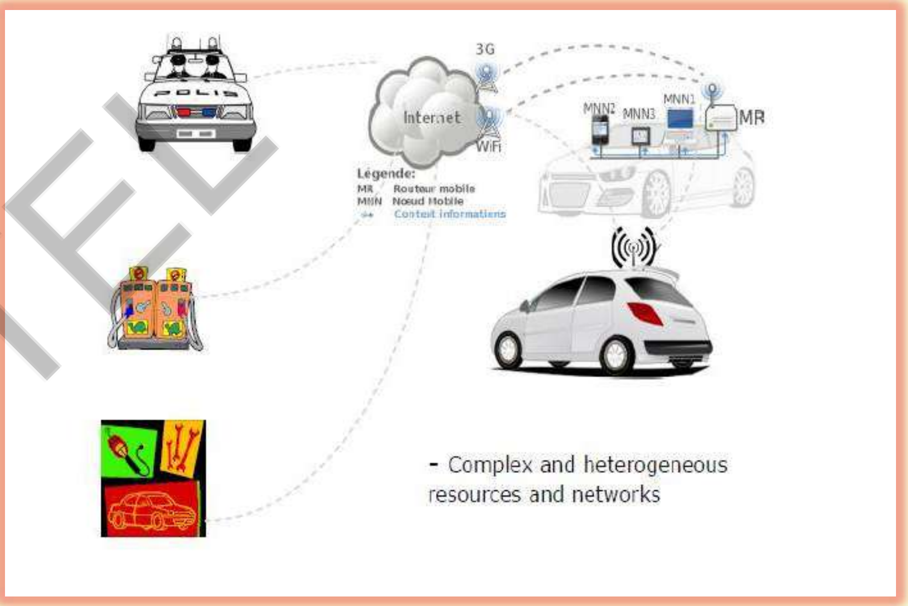
More “*Things*” are being connected!

- Home/daily-life devices
- Business
- Public infrastructure
- Health-care and so on…

Any time, Any place connectivity for Anyone and Anything!



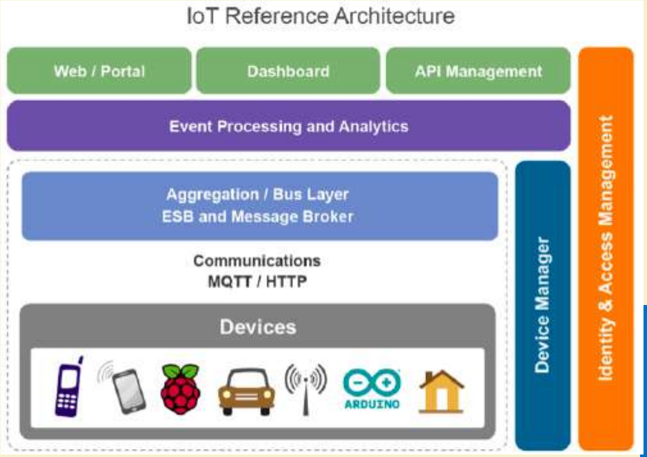
"People" Connecting to "Things"!



- Complex and heterogeneous resources and networks

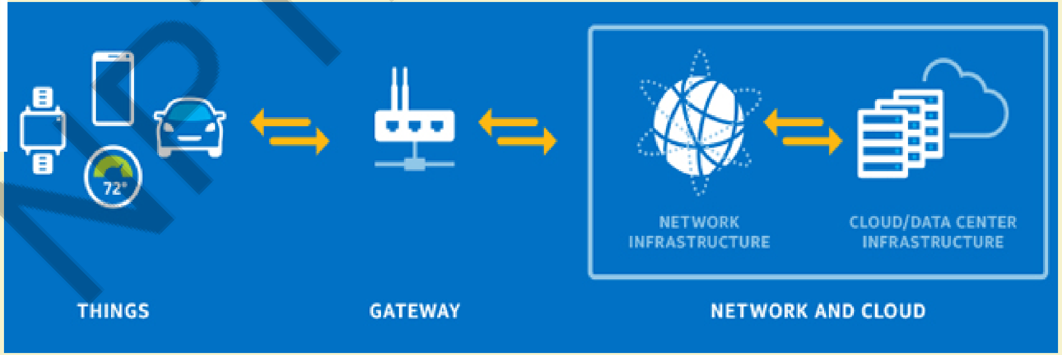
"Things" Connecting to "Things"!

Basic IoT Architecture



An IoT platform has basically three building blocks

- Things
- Gateway
- Network and Cloud



Several Aspects of IoT systems!

- ***Scalability:*** Scale for IoT system applies in terms of the numbers of sensors and actuators connected to the system, in terms of the networks which connect them together, in terms of the amount of data associated with the system and its speed of movement and also in terms of the amount of processing power required.
- ***Big Data***: Many more advanced IoT systems depend on the analysis of vast quantities of data. There is a need, for example, to extract patterns from historical data that can be used to drive decisions about future actions. IoT systems are thus often classic examples of “Big Data” processing.
- ***Role of Cloud computing***: IoT systems frequently involve the use of cloud computing platforms. Cloud computing platforms offer the potential to use large amounts of resources, both in terms of the storage of data and also in the ability to bring flexible and scalable processing resources to the analysis of data. IoT systems are likely to require the use of a variety of processing software – and the adaptability of cloud services is likely to be required in order to deal with new requirements, firmware or system updates and offer new capabilities over time.

Several Aspects of IoT systems (contd…)

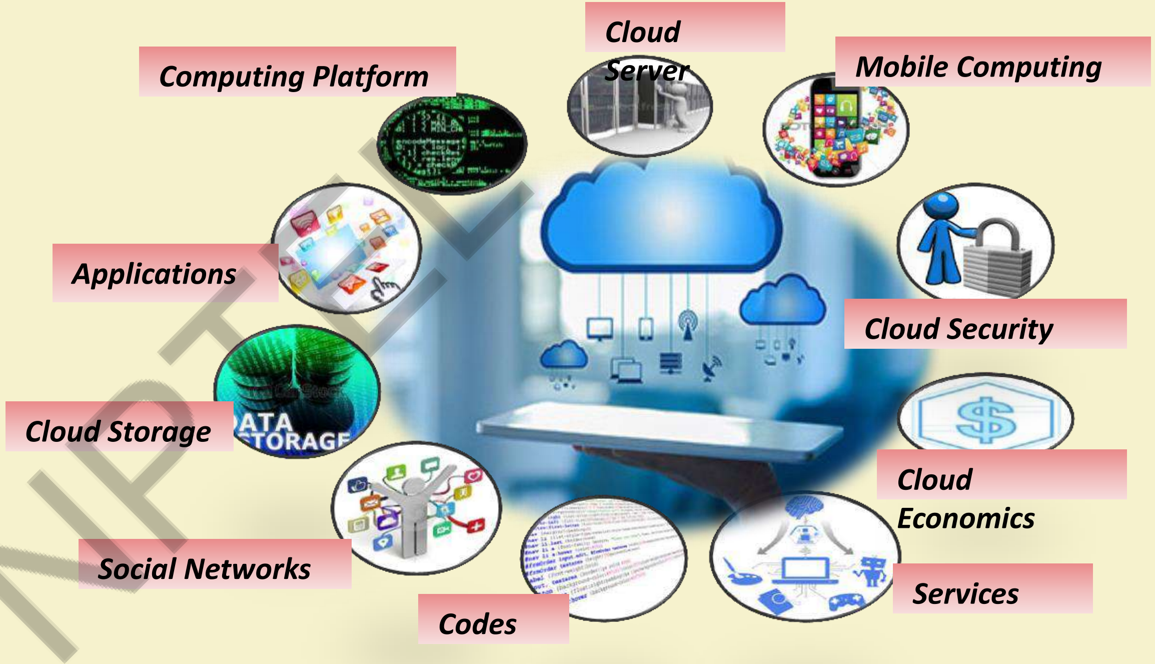
- ***Real time:*** IoT systems often function in real time; data flows in continually about events in progress and there can be a need to produce timely responses to that stream of events.
- ***Highly distributed***: IoT systems can span whole buildings, span whole cities, and even span the globe. Wide distribution can also apply to data – which can be stored at the edge of the network or stored centrally. Distribution can also apply to processing – some processing takes place centrally (in cloud services), but processing can take place at the edge of the network, either in the IoT gateways or even within (more capable types of) sensors and actuators. Today there are officially more mobile devices than people in the world. Mobile devices and networks are one of the best known IoT devices and networks.
- ***Heterogeneous systems***: IoT systems are often built using a very heterogeneous set of. This applies to the sensors and actuators, but also applies to the types of networks involved and the variety of processing components. It is common for sensors to be low-power devices, and it is often the case that these devices use specialized local networks to communicate. To enable internet scale access to devices of this kind, an IoT gateway is used

Cloud Computing!

- Cloud computing enables companies and applications, which are system infrastructure dependent, to be infrastructure-less.

- Cloud infrastructure offers "**pay-as-used** and **on-demand**" services

- Clients can offload their data and applications on cloud for storage and processing

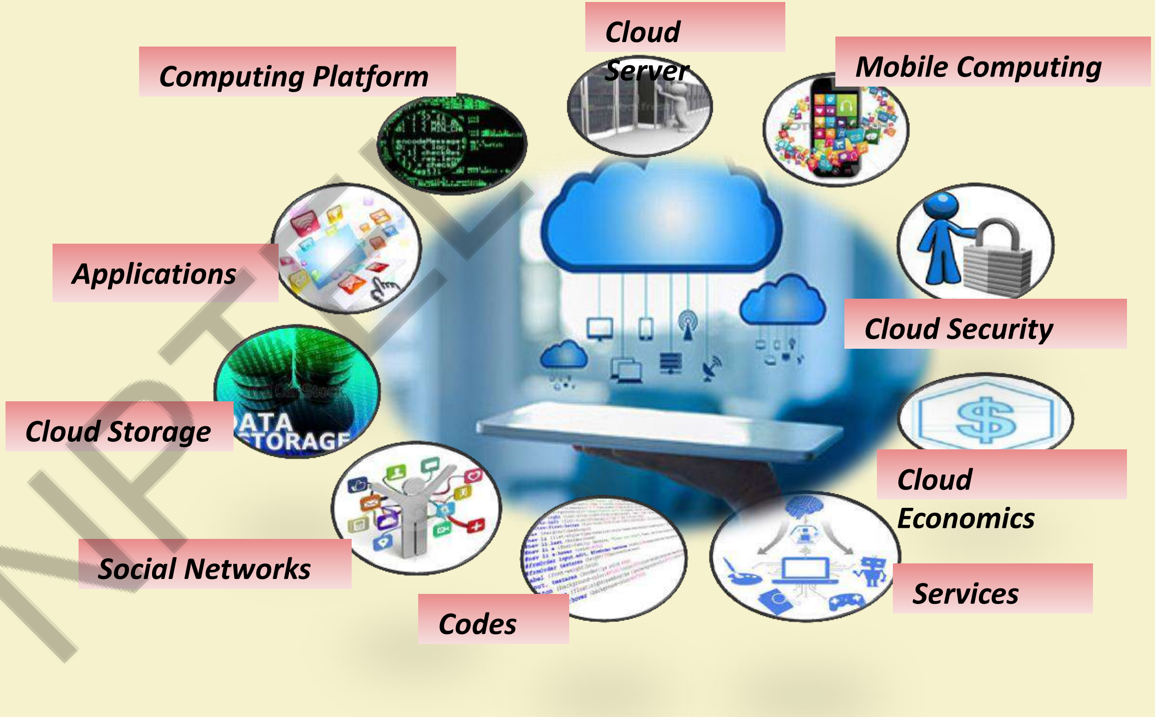


Cloud Computing!

- It enables services to be used without any understanding of the infrastructure.

- Cloud computing works using economies of scale

- Data and services are stored remotely but accessible from "anywhere".

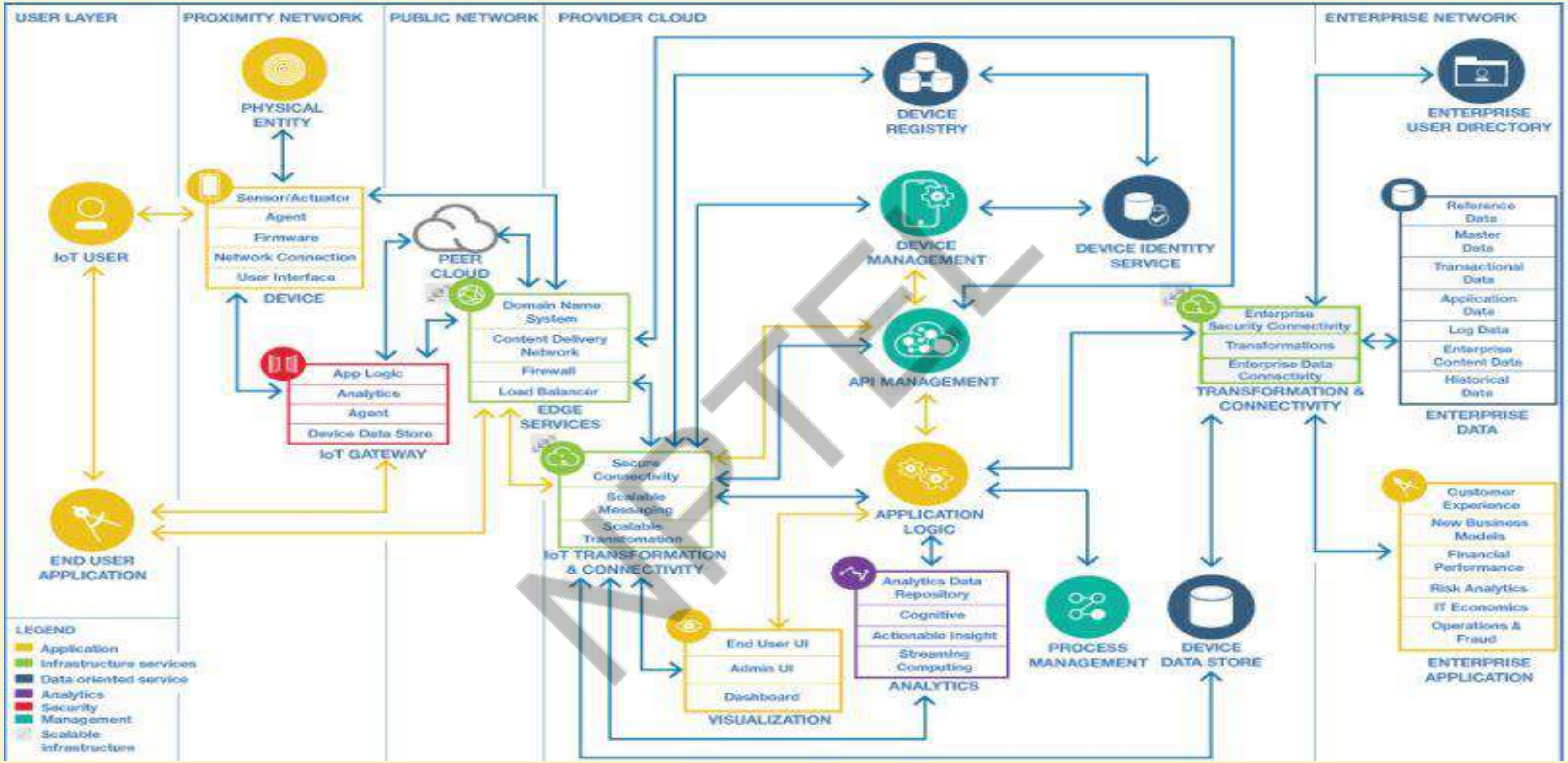


IoT Cloud Systems?

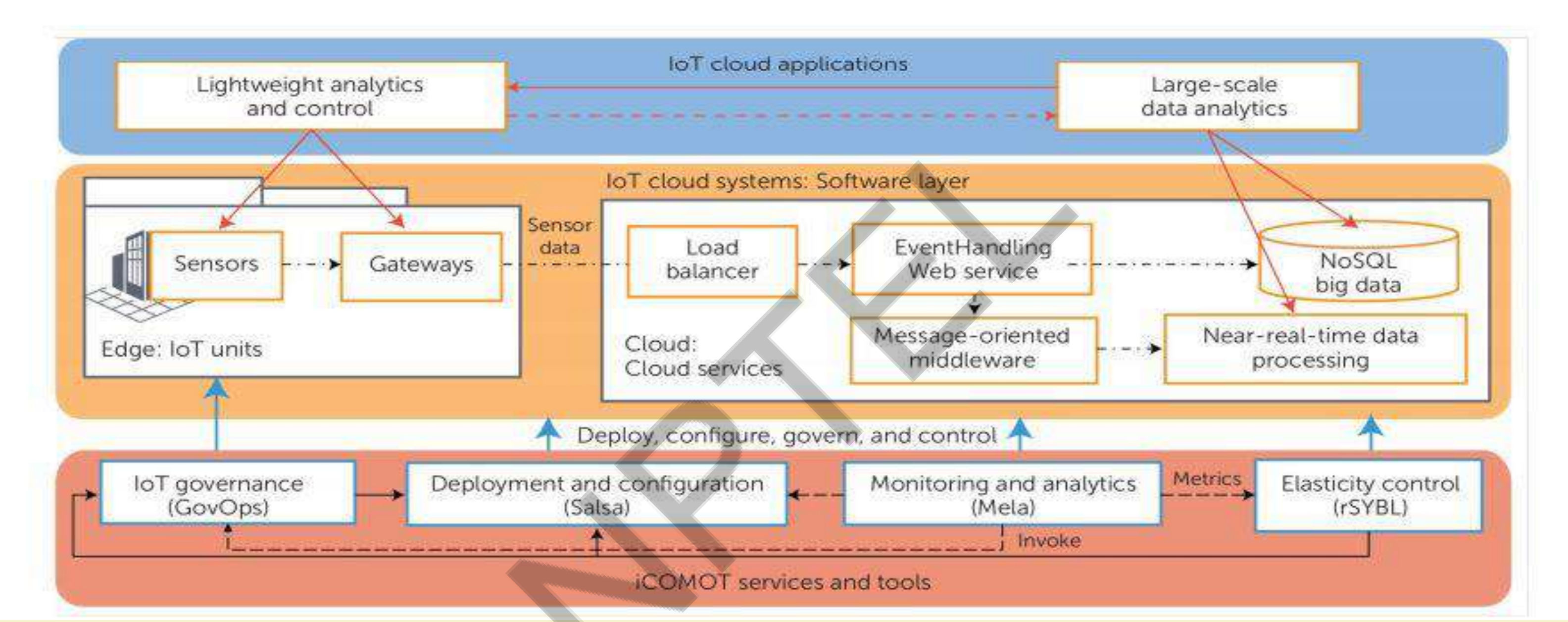
- Recently, there is a *wide adoption and deployment of Internet of Things (IoT)* infrastructures and systems for *various crucial applications*, such as logistics, smart cities, and healthcare. Thi *vices* in clo d cloud se
- Cl gement ca gement te
- Th well as management platforms for IoT. From a high level view, IoT appears to be well integrated with cloud data centers to establish a uniform infrastructure for IoT Cloud applications

An integration between IoT and cloud services allows coordination among IoT and cloud services. That is, a cloud service can request an IoT service, which includes several IoT elements, to reduce the amount of sensing data or the IoT service can request cloud services to provision more resources for future incoming data

Cloud Components for IoT



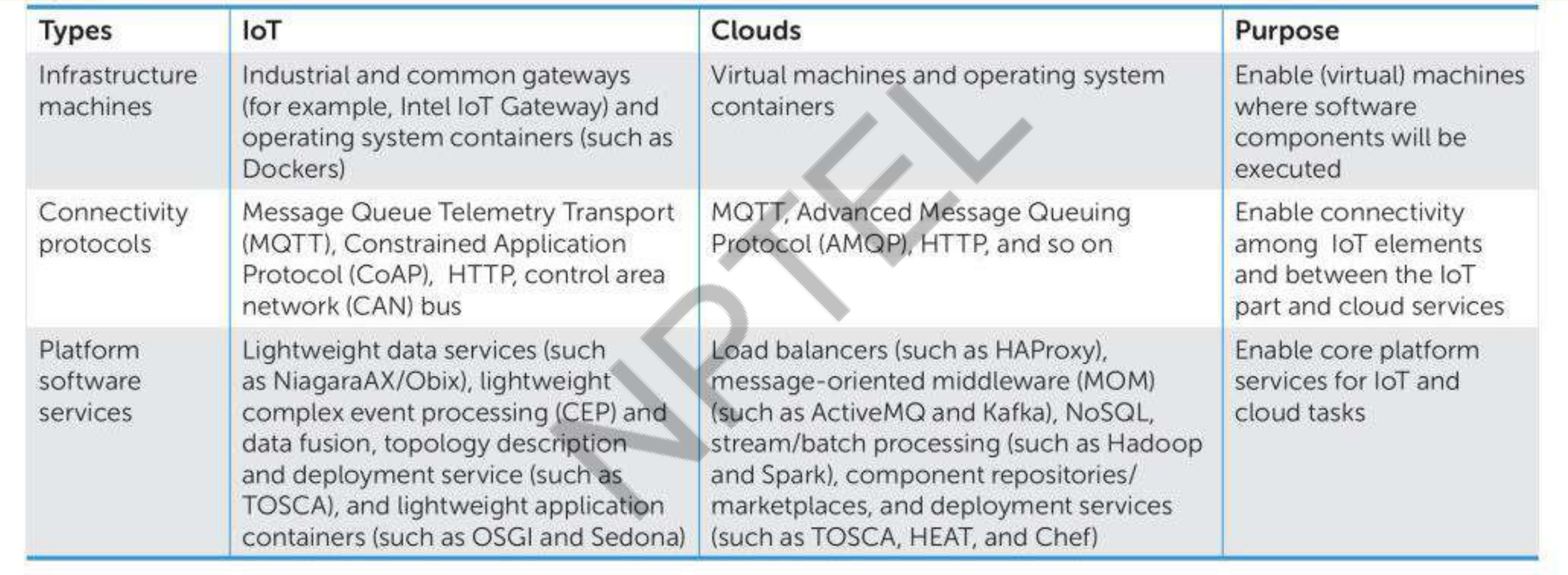
iCOMOT: An IoT Cloud System


IoT cloud applications
Lightweight analytics and control
Large-scale data analytics
IoT cloud systems: Software layer
Sensors
Gateways
Edge: IoT units
Sensor data
Load balancer
EventHandling Web service
NoSQL big data
Message-oriented middleware
Near-real-time data processing
Cloud: Cloud services
Deploy, configure, govern, and control
IoT governance (GovOps)
Deployment and configuration (Salsa)
Monitoring and analytics (Mela)
Metrics
Elasticity control (rSYBL)
Invoke
iCOMOT services and tools

Top layer represents typical IoT applications executed across IoT and Clouds.
Middle layer represents the software layer as an IoT cloud system built on top of various types of cloud services and IoT elements.
Bottom layer shows different tools and services from iCOMOT that can be used to monitor, control, and configure the software layer.

H.-L. Truong et al., “iCOMOT: Toolset for Managing IoT Cloud Systems,”
Demo, 16th IEEE Int’l Conf. Mobile Data Management, 2015

Infrastructure, Protocols and **Software Platforms** for establishing an **Internet of Things (IoT) Cloud system**



Types	IoT	Clouds	Purpose
Infrastructure machines	Industrial and common gateways (for example, Intel IoT Gateway) and operating system containers (such as Dockers)	Virtual machines and operating system containers	Enable (virtual) machines where software components will be executed
Connectivity protocols	Message Queue Telemetry Transport (MQTT), Constrained Application Protocol (CoAP), HTTP, control area network (CAN) bus	MQTT, Advanced Message Queuing Protocol (AMQP), HTTP, and so on	Enable connectivity among IoT elements and between the IoT part and cloud services
Platform software services	Lightweight data services (such as NiagaraAX/Obix), lightweight complex event processing (CEP) and data fusion, topology description and deployment service (such as TOSCA), and lightweight application containers (such as OSGI and Sedona)	Load balancers (such as HAProxy), message-oriented middleware (MOM) (such as ActiveMQ and Kafka), NoSQL, stream/batch processing (such as Hadoop and Spark), component repositories/ marketplaces, and deployment services (such as TOSCA, HEAT, and Chef)	Enable core platform services for IoT and cloud tasks

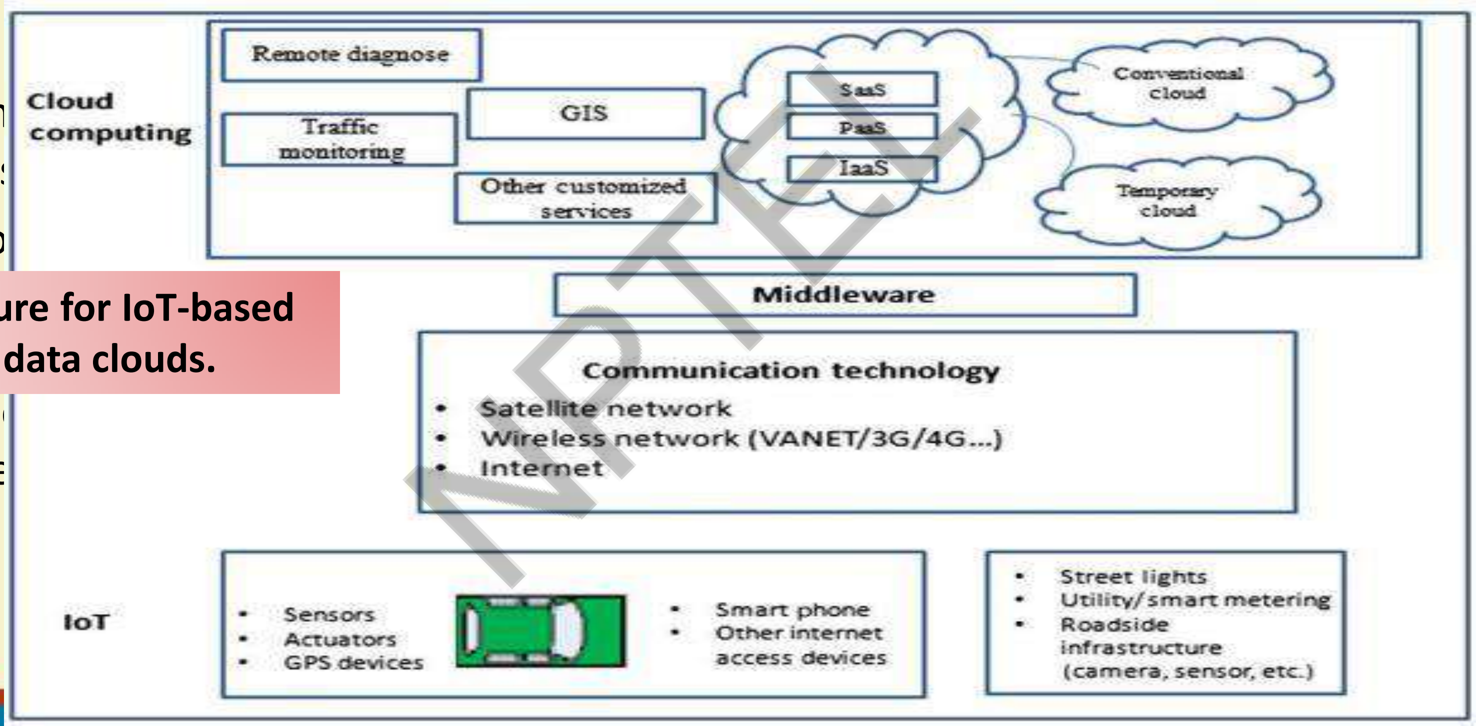
Truong, Hong-Linh, and Schahram Dustdar. Principles for engineering IoT cloud systems." *IEEE Cloud Computing* 2.2 (2015): 68-76.

Motivating example: ***Developing Vehicular Data Cloud Services in the IoT Environment***

- The
  pron
  tran
- A no d

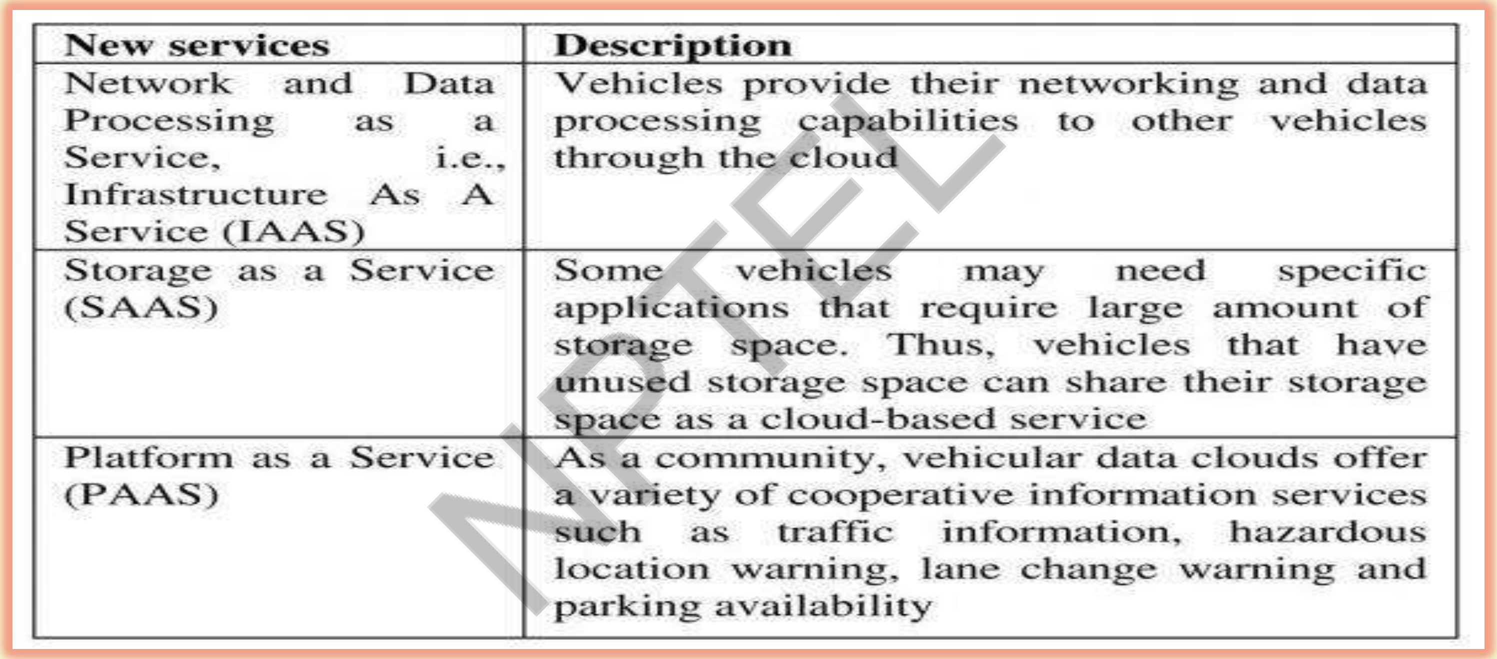
Architecture for IoT-based vehicular data clouds.

vehi e
Baye



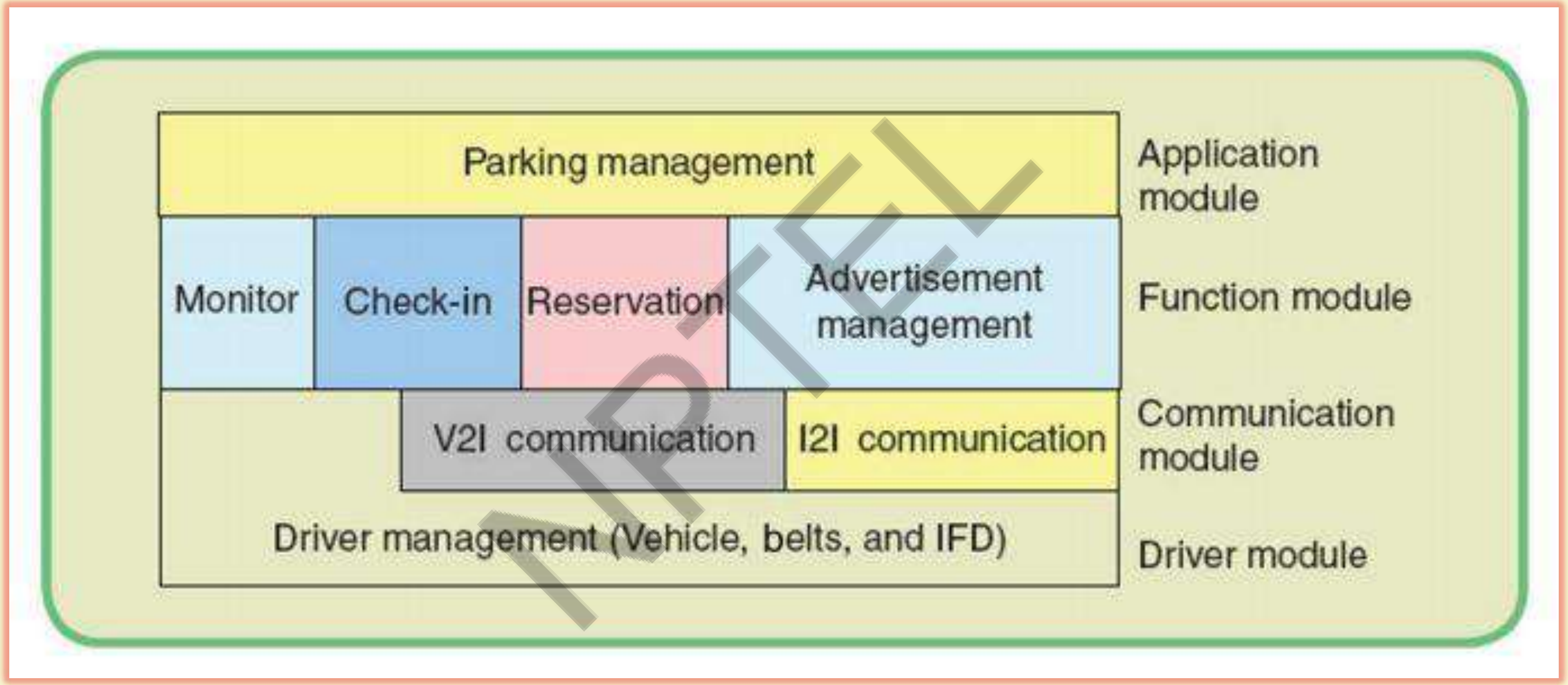
He, Wu, Gongjun Yan, and Li Da Xu. "Developing vehicular data cloud services in the IoT environment." *IEEE Transactions on Industrial Informatics* 10.2 (2014): 1587-1595.

Services for IoT-based Vehicular Data Clouds



New services	Description
Network and Data Processing as a Service, i.e., Infrastructure As A Service (IAAS)	Vehicles provide their networking and data processing capabilities to other vehicles through the cloud
Storage as a Service (SAAS)	Some vehicles may need specific applications that require large amount of storage space. Thus, vehicles that have unused storage space can share their storage space as a cloud-based service
Platform as a Service (PAAS)	As a community, vehicular data clouds offer a variety of cooperative information services such as traffic information, hazardous location warning, lane change warning and parking availability

Architecture for Intelligent Parking Cloud service



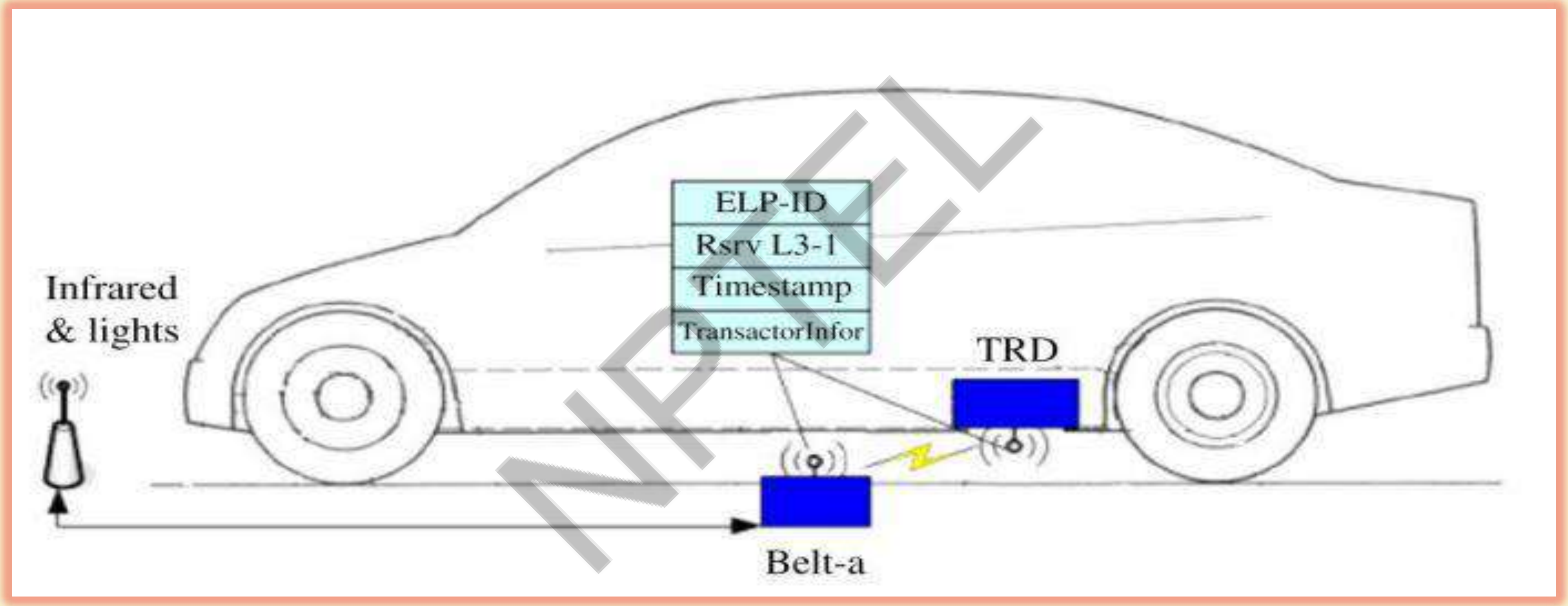
Parking management — Application module

Monitor | Check-in | Reservation | Advertisement management — Function module

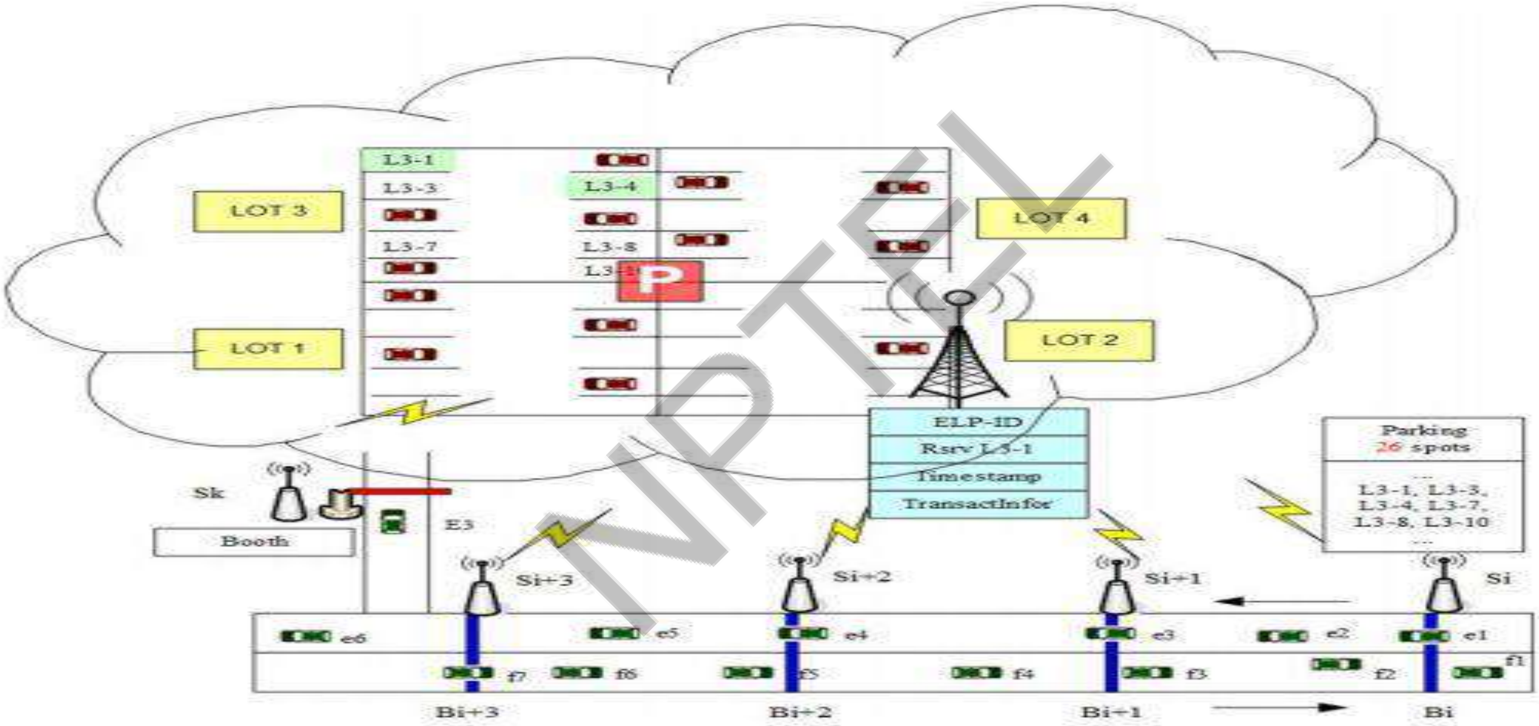
V2I communication | I2I communication — Communication module

Driver management (Vehicle, belts, and IFD) — Driver module

Vacancy detections by Sensors



Parking cloud service



Summary

- Internet of Things (IoT) is a dynamic and exciting area of IT. Many IoT systems are going to be created over the next few years, covering wide variety of areas, like domestic, commercial, industrial, health and government contexts

- IoT systems have several challenges, namely scale, speed, safety, security and privacy

- Cloud computing platforms offer the potential to use large amounts of resources, both in terms of the storage of data and also in the ability to bring flexible and scalable processing resources to the analysis of data

- IoT Cloud Platform is an enabling paradigm to realize variety of services

References

- Cloud Standards Customer Council 2015, Cloud Customer Architecture for Big Data and Analytics, Version 1.1 http://www.cloud-council.org/deliverables/CSCC-Customer-Cloud-Architecture-for-Big-Data-andAnalytics.pdf
- He, Wu, Gongjun Yan, and Li Da Xu. "Developing vehicular data cloud services in the IoT environment." *IEEE Transactions on Industrial Informatics* 10.2 (2014): 1587-1595.
- H.-L. Truong et al., “iCOMOT: Toolset for Managing IoT Cloud Systems,” demo, 16th IEEE Int’l Conf. Mobile Data Management, 2015
- Truong, Hong-Linh, and Schahram Dustdar. Principles for engineering IoT cloud systems." *IEEE Cloud Computing* 2.2 (2015): 68-76

Thank You!

IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

CLOUD COMPUTING

Course Summary and Research Areas

PROF. SOUMYA K. GHOSH

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
IIT KHARAGPUR

Course Summary

- Introduction to Cloud Computing
  - Cloud Computing (NIST Model)
  - Properties, Characteristics & Disadvantages
- Cloud Computing Architecture
  - Cloud computing stack
  - Service Models (XaaS)
  - Deployment Models
- Service Management in Cloud Computing
  - Service Level Agreements(SLAs)
  - Cloud Economics
- Resource Management in Cloud

Course Summary (contd.)

- Data Management in Cloud Computing
    - Data, Scalability & Cloud Services
    - Database & Data Stores in Cloud
    - GFS, HDFS, Map-Reduce paradigm
- Cloud Security
    - Identity & Access Management
    - Access Control
    - Trust, Reputation, Risk
    - Authentication in cloud computing
- Case Study on Open Source and Commercial Clouds
- Research trend - Fog Computing, Sensor Cloud, Container Technology, Green Cloud etc.

Cloud Computing – Research Areas

Cloud Infrastructure and Services

- Cloud Computing Architectures
- Storage ad Data Architectures
- Distributed and Cloud Networking
- Infrastructure Technologies

- IaaS, PaaS, SaaS
- Storage-as-a-Service
- Network-as-a-Service
- Information-as-a-Service

Cloud Management, Operations and Monitoring

- Cloud Composition, Service Orchestration
- Cloud Federation, Bridging, and Bursting
- Cloud Migration
- Hybrid Cloud Integration
- **Green and Energy Management of Cloud Computing**
- Configuration and Capacity Management
- Cloud Workload Profiling and Deployment Control
- Cloud Metering, Monitoring, Auditing
- Service Management

Cloud Security

- Data Privacy
- Access Control
- Identity Management
- Side Channel Attacks
- Security-as-a-Service

Performance, Scalability, Reliability

- Performance of cloud systems and Applications
- Cloud Availability and Reliability
- Micro-services based architecture

Systems Software and Hardware

- Virtualization Technology
- Service Composition
- Cloud Provisioning Orchestration
- Hardware Architecture support for Cloud Computing

Data Analytics in Cloud

- Analytics Applications
- Scientific Computing and Data Management
- Big data management and analytics
- Storage, Data, and Analytics Clouds

Cloud Computing – Service Management

- Services Discovery and Recommendation
- Services Composition
- Services QoS Management
- Services Security and Privacy
- Semantic Services
- Service Oriented Software Engineering

Cloud and Other Technologies

- Fog Computing
- IoT Cloud
- Container Technology

Thank You!