A Taxonomy and Survey of Energy-Efficient Data Centers and Cloud Computing Systems

Anton Beloglazov, Rajkumar Buyya, Young Choon Lee, and Albert Zomaya

Present by Leping Wang
1/25/2012

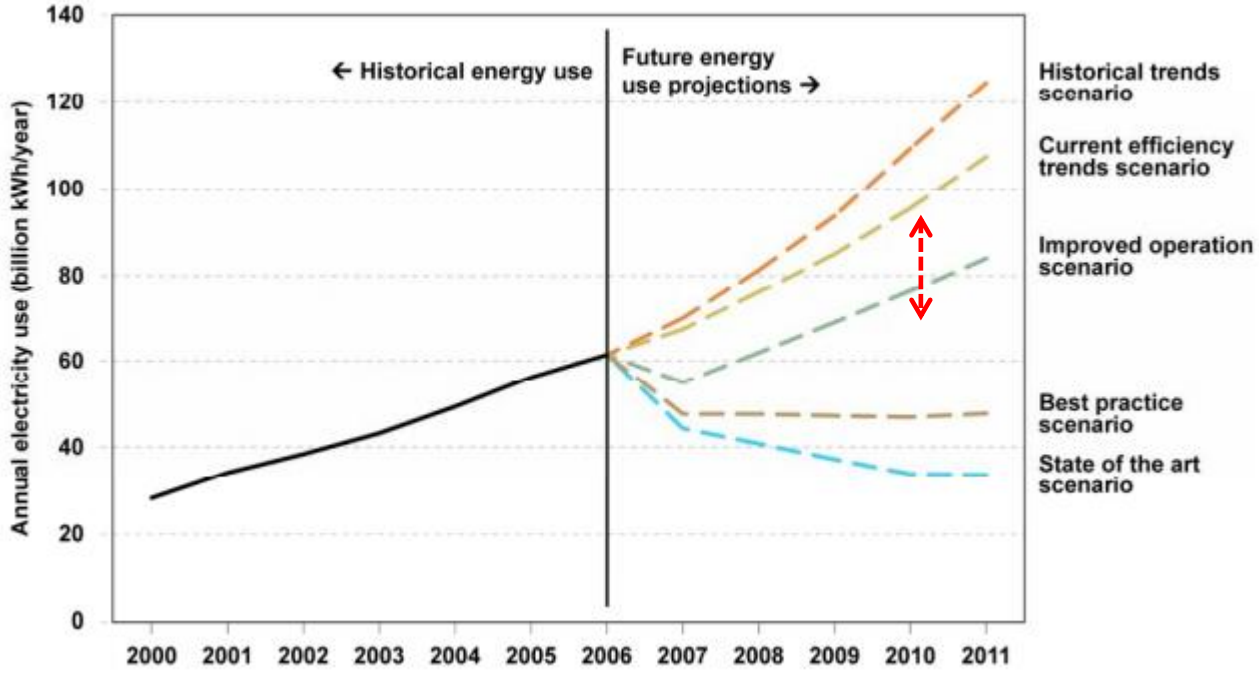
Outline

- **Background**
- **Models of Power Consumption**
- **Taxonomy of PM in Computing System**
- **PM Techniques in a Single Server**
- **PM Techniques in Data Centers**
- **Future Work**

Background

- Increasing Electricity Bill
  - The cost of energy consumed by a server during its life time may exceed the hardware cost. And the problem is even worse for clusters and data centers.

Figure ES-1. Comparison of Projected Electricity Use, All Scenarios, 2007 to 2011



U.S. data center energy use could double to more than 120 billion kWh from 2006 to 2011, equal to annual electricity costs of $7.4 billion, accounted for 2% of all electricity

Background

- Increasing carbon dioxide(CO_2) Emissions
  - Annual source energy use of a 2MW data center is equal to the amount of energy consumed by 4,600 typical U.S. cars in one year.

=

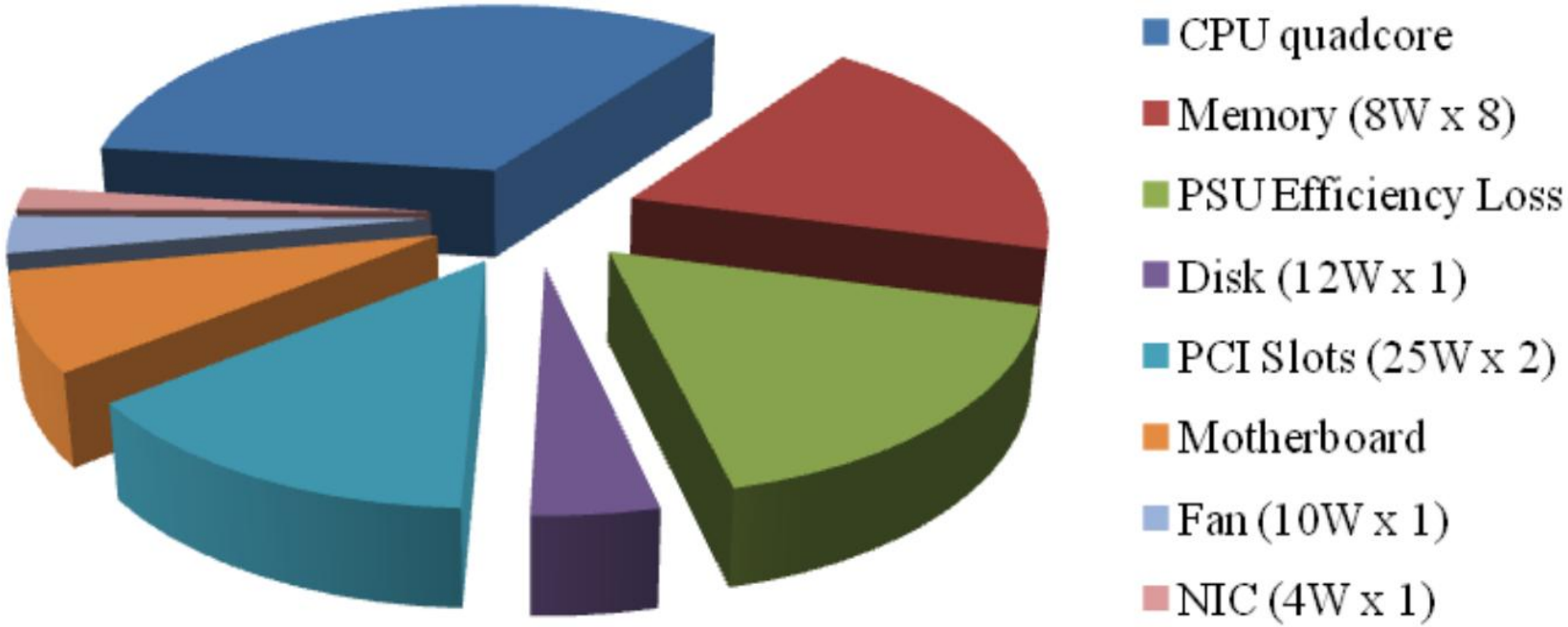
Background

- Power-efficiency (performance per watt, green IT, green computing, eco computing) now becomes a first-order concern for designers of modern computing system.

Models of Power Consumption

- Where does the power go?

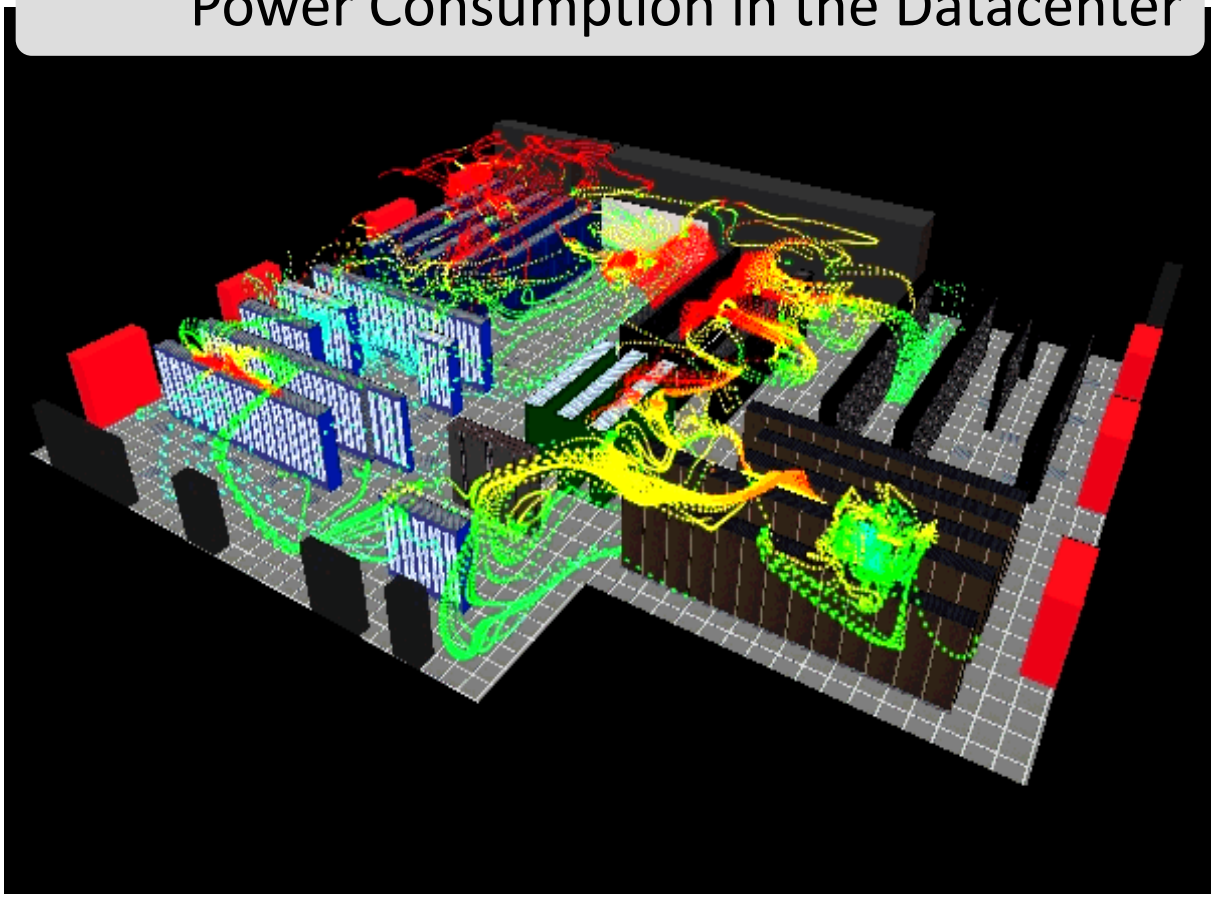
Power Consumption in one Server



Models of Power Consumption

- Where does the power go?

Power Consumption in the Datacenter

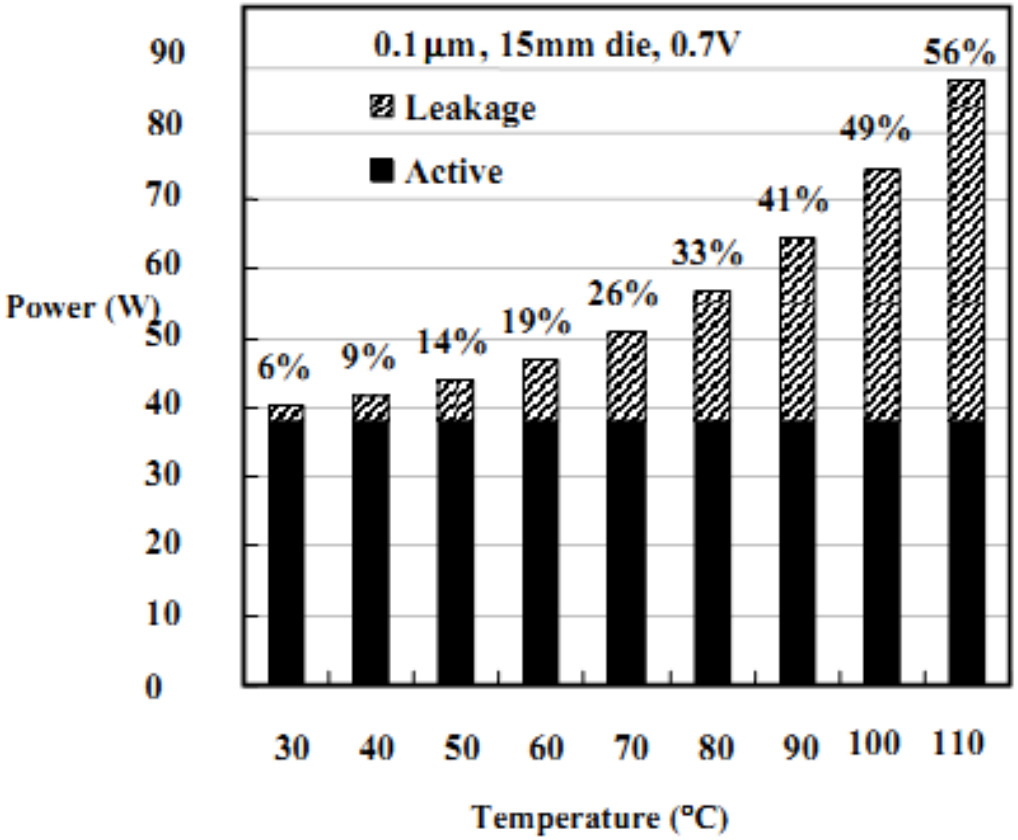


Server/Storage	**50%**
Computer Rm. AC	**34%**
Power Conversion	**7%**
Network	**7%**
Lighting	**2%**

Compute resources and particularly **servers** are at the heart of a complex, evolving system!

Models of Power Consumption

- Static Power Consumption
  - Static Power Consumption is independent of clock rates, device usage scenarios and system status, for example, the leakage power consumption.



On a typical ASIC in a modern nanometer process, the leakage power consumption cannot keep independent . It is very related to the temperature. This paper does not mention this problem.

Models of Power Consumption

- Dynamic Power Consumption
  - Created by circuit activity(transistor switches, changes of values in register, etc)
  - Depends mainly on system's status (usage scenario, clock rates and IO activity)
  - Defined as follows:

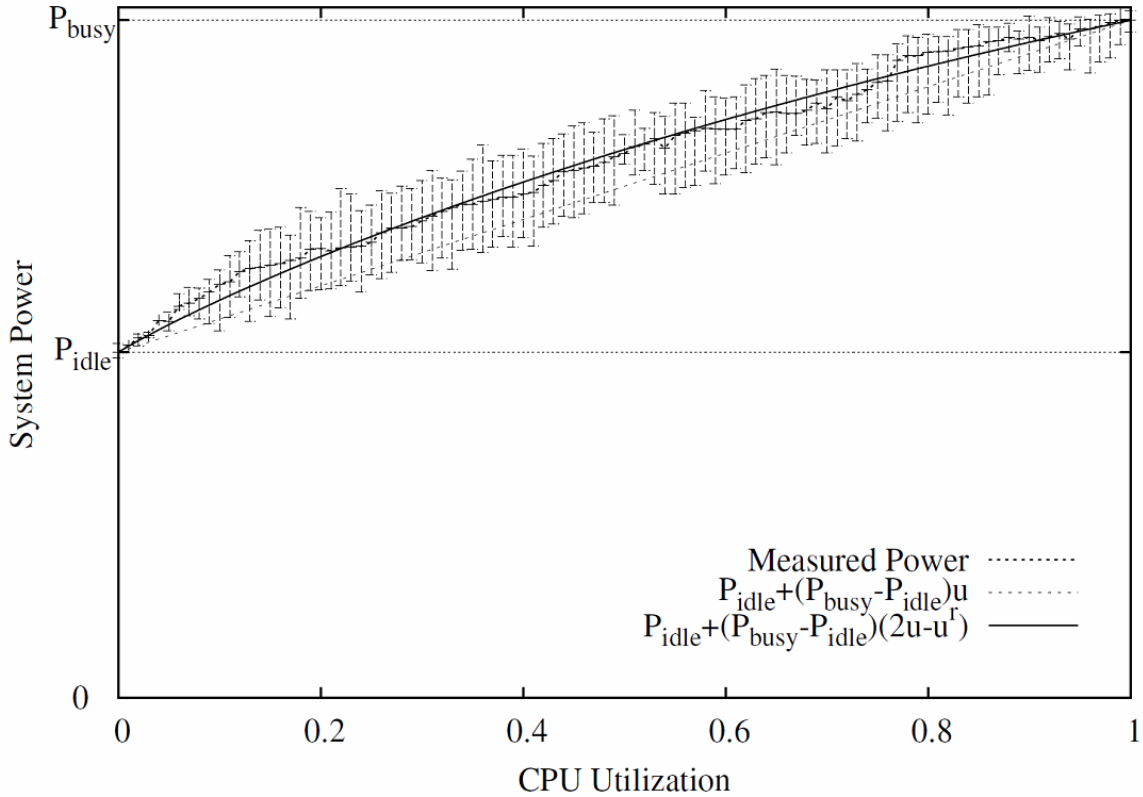
$$P_{dynamic} = a \cdot C \cdot V^2 \cdot f$$

a: switching activity
C: equivalent capacitance
V: supply voltage
f: clock frequency

Models of Power Consumption

- Modeling Power Consumption
  - Paper [9] presented a strong relationship between the CPU utilization and a server total power consumption.

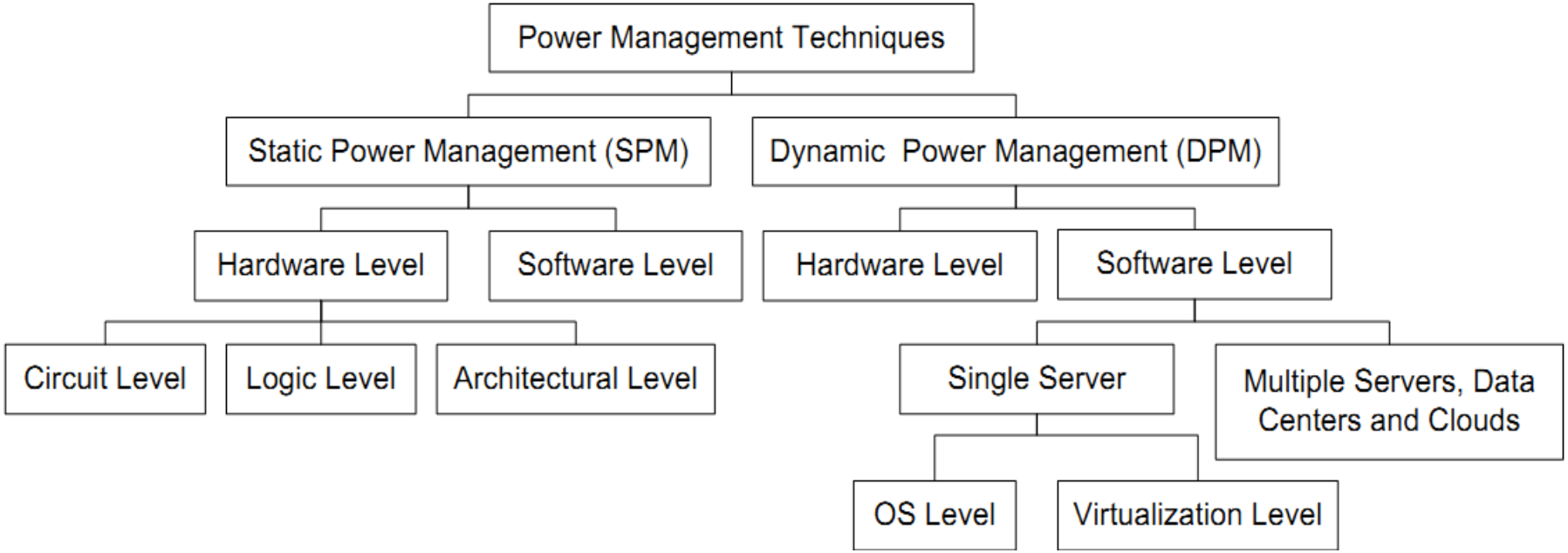
$$P(u) = P_{idle} + (P_{busy} - P_{idle}) \cdot (2u - u^r)$$



u: CPU's utilization
r: calibration parameter calculated practically

Taxonomy of PM in Computing System

- High level taxonomy of PM



  - SPM mainly focuses on optimizing the circuit, logic and architecture of the system at the design time.
  - This paper focuses on DPM that include methods and strategies for run-time adaptation of a system's behavior according to current resource requirements or any other dynamic characteristic of the system's state.

PM in a Single Server

- The key issue of PM is how to increase the system's utilization:

  – Method 1:
  Deactivate computer's Components(DCD)
  Related research topics:
    - Determine time thresholds of idle(inactive) periods considering the transition overhead of components
    - Adaptive and predictive methods

PM in a Single Server

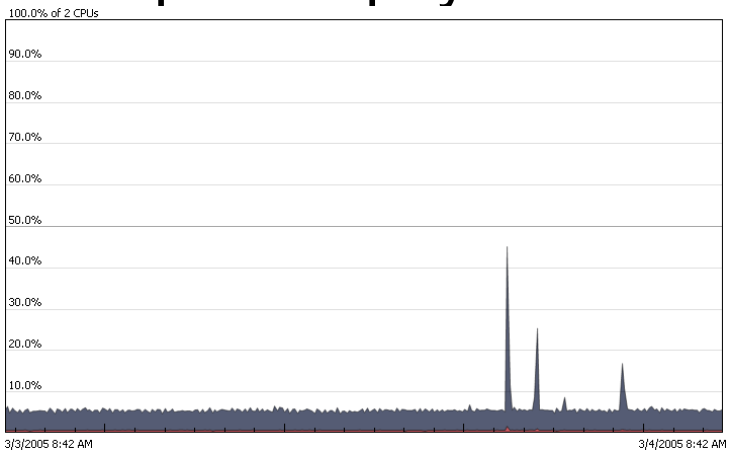
- The key issue of PM is how to increase the system's utilization:

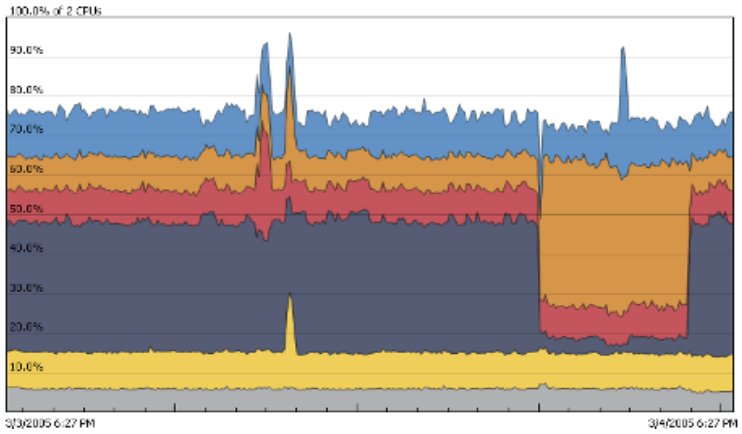
  – Method 2:
  Decrease computer's performance (DVFS)
  Related research topics:
    - Real-time tasks with DVFS
    - QoS and DVFS

PM in a Single Server

- The key issue of PM is how to increase the system's utilization:
  - Method 3: Multiplex computer's physical resources (Virtualization)

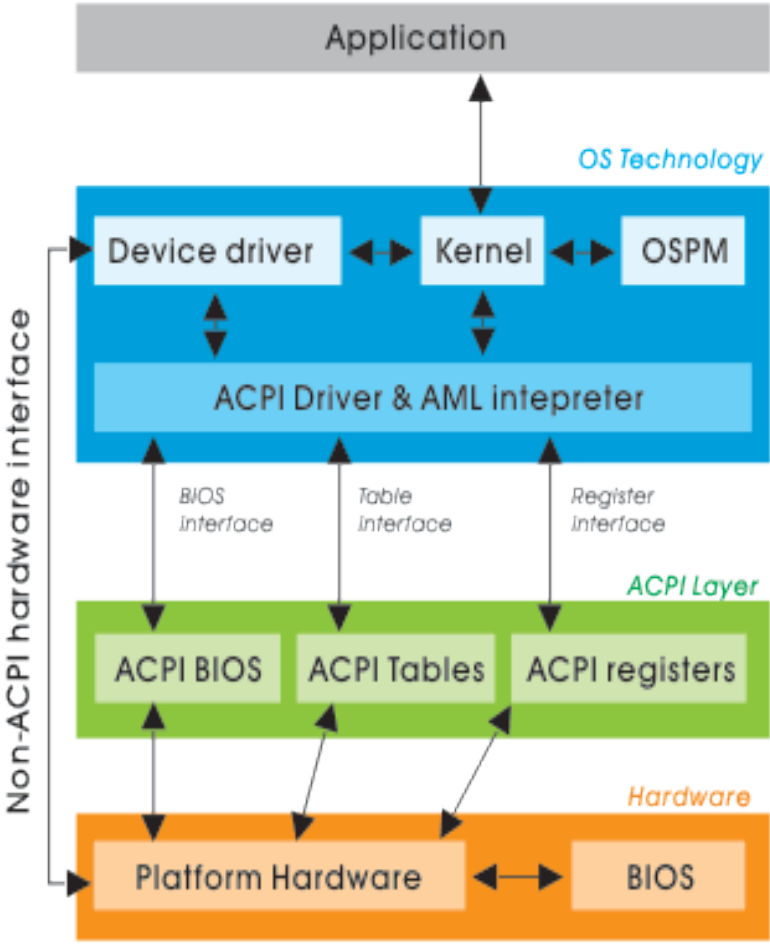
Virtualization





PM in a Single Server

- OS support
  - Advanced Configuration and Power Interface (ACPI)
    - Defines platform-independent interface for hardware discovery, configuration, power management and monitoring.
    - Motherboard, CPU, NIC, Power Supply… interact with ACPI-compliant OS through AML(ACPI machine Language)
    - Global states, Device states, CPU states, Performance states



PM in a Single Server

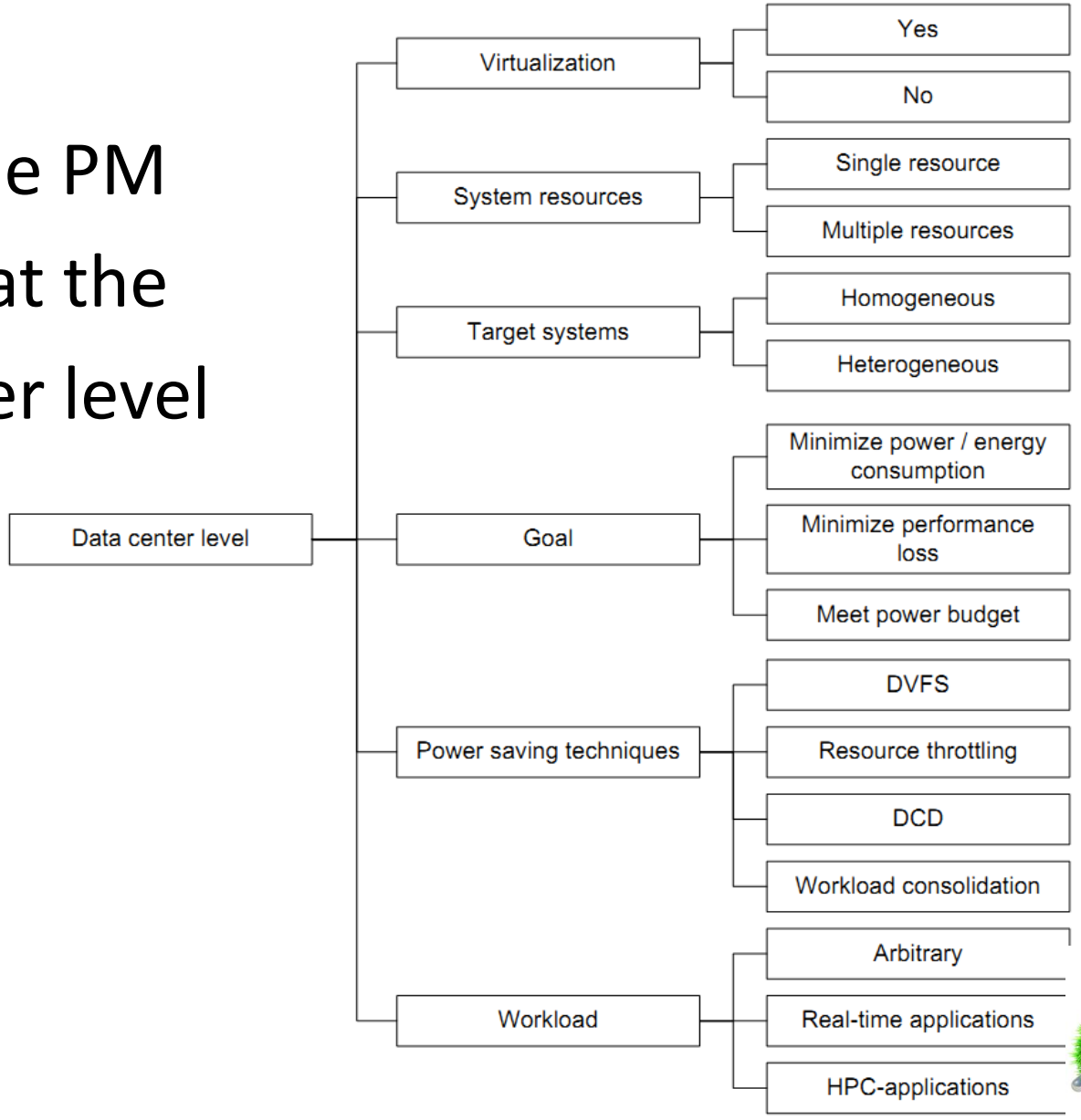
- OS support

  ACPI does not define any PM policies but the interface
  - Windows
    - PM API and applications
  - Linux Kernel
    - Paper [18] developed an in-kernel real-time power manage for Linux called the ondemand governor
    - Keeps the CPU 80% busy by adaptively setting a clock frequency and voltage pair
  - Xen
    - Four governors:
      - Ondemand
      - Userspace
      - Performance
      - Powersave

PM in Data Centers

PM in Data Centers

- Classify the PM methods at the data center level



PM in Data Centers

- PM Techniques in Data Centers
  - Turn off or hibernate idle servers
  - Dynamically scale operating frequency/voltage (DVFS) for underutilized servers
  - VM consolidation
  - Keep servers running at their power-efficient state
  - Distribute more requests to power-efficient servers

PM in Data Centers

- Load Management for Power and Performance in Clusters
  - Homogeneous cluster
  - Server power switching is only used
  - Relatively small difference in power consumption between an idle node(70W) and a fully utilized node(94W). So, less servers always save more power when handling same workload
  - Predict the workload and performance up/degradation by keeping tack of the demand for sources
  - The acceptable performance degradation(QoS) is specified by users.
  - Activate as few servers as possible

PM in Data Centers

- Energy-Efficient Server Clusters
  - Homogeneous clusters
  - Even workload distribution
  - Vary-ON/Off mechanism
  - Independent DVFS(IVS)
    Coordinated DVFS(CVS)
  - N or N+1 servers to process current workload?

PM in Data Centers

- Energy-Efficient Server Clusters
  - N or N - 1 servers to process current workload?

$$n \times P(f_1) = n \times \left(c_0 + c_1 f_1^3\right)$$

Power consumption of N server

$$(n-1) \times P\left(\frac{n}{n-1} f_1\right) = (n-1) \times \left(c_0 + c_1 \left(\frac{n}{n-1} f_1\right)^3\right)$$

Power consumption of N-1 server

Solve this equation

$$(n-1) \times \left(c_0 + c_1 \left(\frac{n}{n-1} f_1\right)^3\right) < n \times \left(c_0 + c_1 f_1^3\right)$$

When is N server's power consumption Greater than N-1 server's

$$f_{varyoff}(n) = \sqrt[3]{\frac{c_0}{c_1} \frac{(n-1)^2}{2n^2 - n}}$$

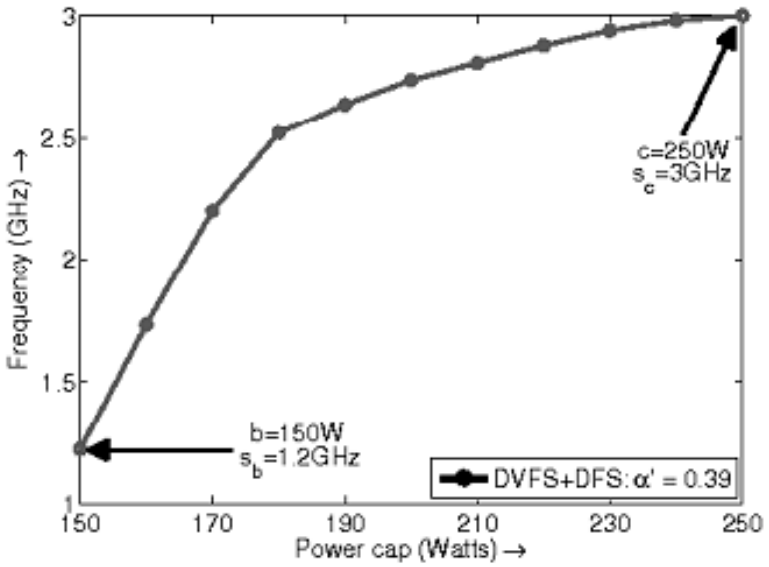
When servers' frequency decrease to $f_{varyoff}$ (n), one server should be turned off.

$$fvaryon(n) = \sqrt[3]{\frac{c_0}{c_1} \frac{(n+1)^2}{2n^2 + n}}$$

When servers' frequency increase to f_{varyon} (n), one more server should be turned on.

PM in Data Centers

- Optimal Power Allocation in Server Farms
  - Given power budget, get optimal performance
  - Experimentally approximate(curve fit) the CPU frequency and power allocation



$$s = s_b + \alpha' \sqrt[3]{\mathcal{P} - b}.$$

s: CPU frequency

s_b: The frequency of a fully utilized server running at *b* Watts

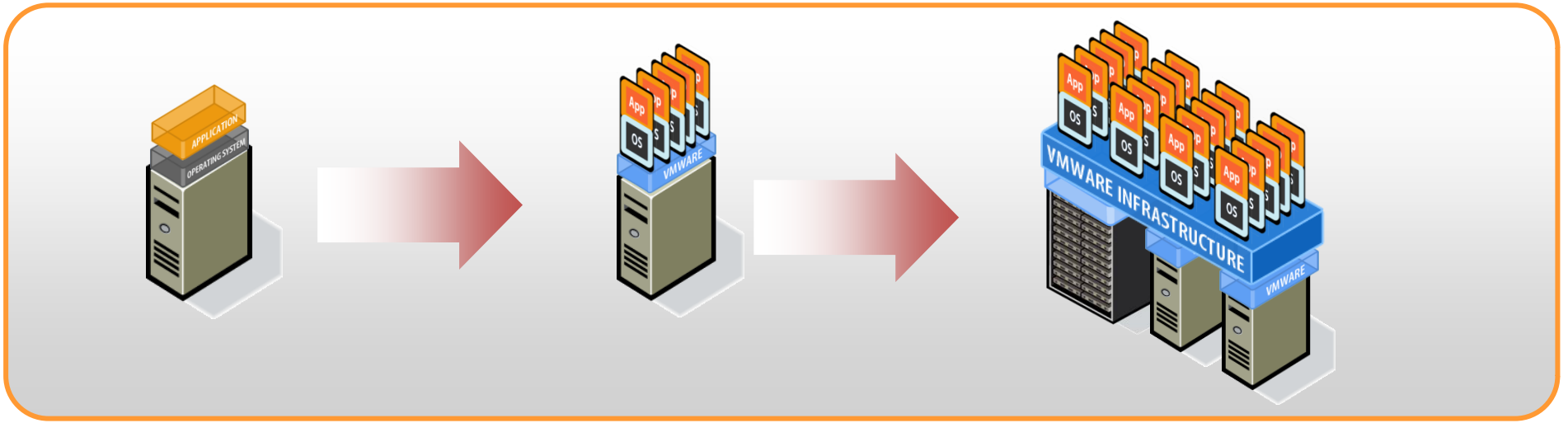
a: the practical coefficient

P: Power allocation

B: The minimum power consumed by a fully-utilized server running at its minimum frequency

PM in Data Centers

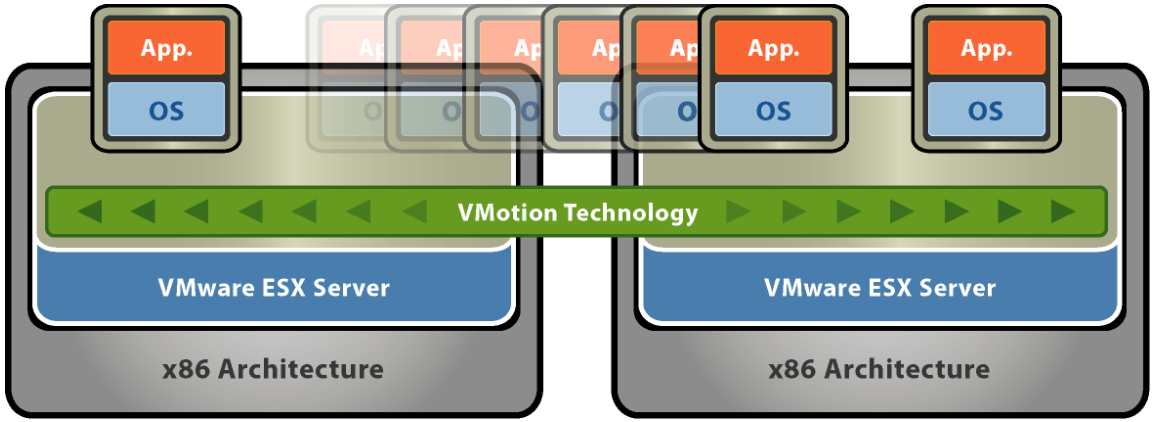
- PM in virtualized Data Centers:



PM in Data Centers

- PM in virtualized Data Centers:
  - Multiple QoS requirements
  - Resource allocation between VMs
  - DVFS may affect all VMs hosted in one server
  - Live Migration

Let's move live, running virtual machines from one host to another while maintaining continuous service availability.



PM in Data Centers

- VirtualPower: Coordinated Power Management (2007)
  - Soft scale down (providing a VM less time for utilizing the resource)VM's performance, increase the idle time of a server to save power
  - The global policies use knowledge of rack- or blade-level characteristics and requirements to consolidate VMs using migration. Then hibernate idle physical servers to save power

PM in Data Centers

- Power-aware Provisioning of Cloud Resources for Real-time Services
  - Goal: Provide **real-time services** on virtualized servers of the data center, while **maximizing profits (money)** and **reducing energy consumption** as much as possible
  - Method: **adaptive-DVFS/proportional sharing scheduling**
    - DVFS: Affect the performance of all VMs hosted in one server
    - PSC: Adjust the resource sharing among VMs hosted in one server, fine-tune any VM's performance

PM in Data Centers

- Energy model

$$E = \alpha \cdot t \cdot S^2$$

a: fixed practical coefficient
t: execution time
S: processor speed (Million Instructions per second)

- RT-VM(Real-time Virtual Machine) V_i has 3 parameters:

$$V_i(u_i, m_i, d_i)$$

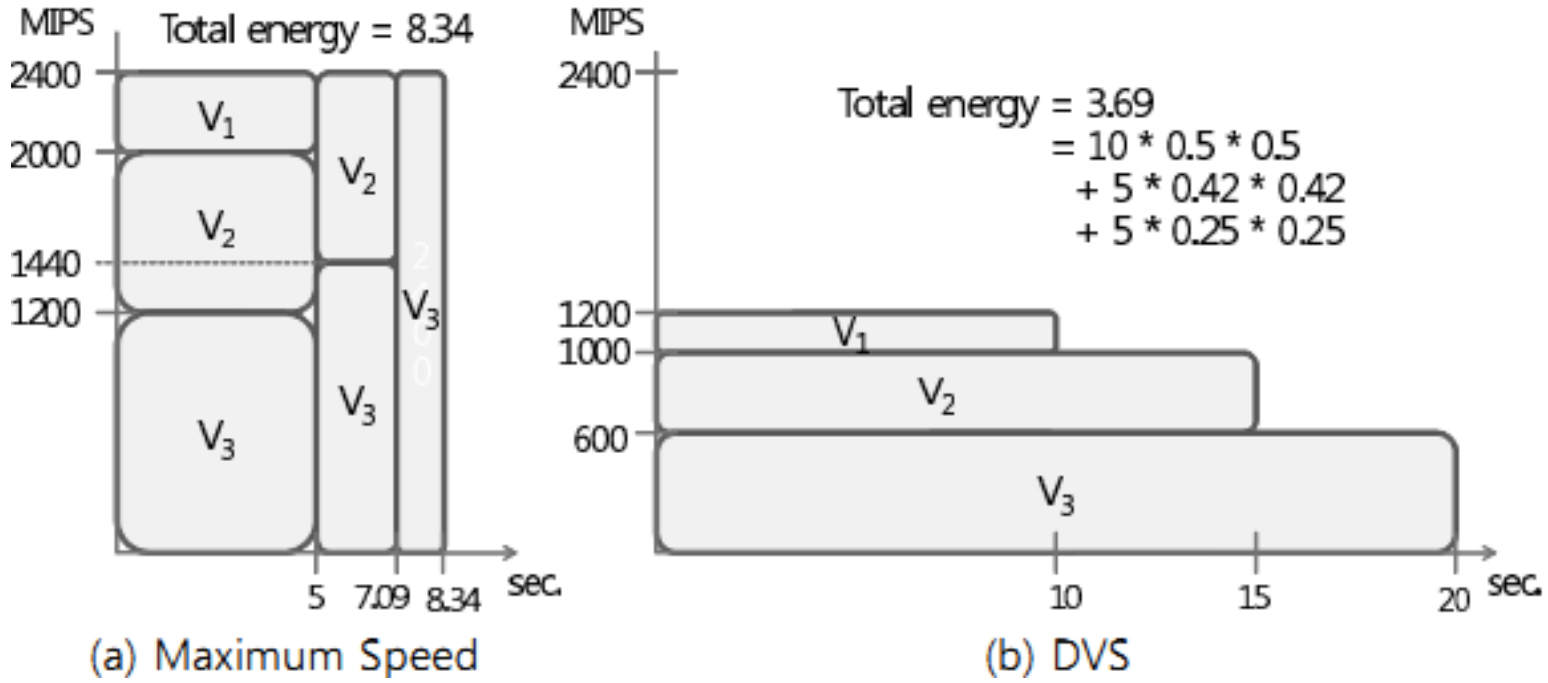
- u_i : utilization of real-time applications
- m_i : MIPS (Million Instructions Per Second) rate of the based virtual machine
- d_i : lifetime or deadline

For example, V1(0.2, 1000, 10) requires the utilization 20% on 1000-MIPS machine by the deadline 10 sec.

PM in Data Centers

– One example about DFVS and proportional sharing scheduling

V1(0.2, 1000, 10), V2(0.8, 500, 15) , V3(0.5, 1200, 20)



PM in Data Centers

- Profit consideration
  - Operation in higher processor speed can accept more RT-VMs to increase datacenters' profit; Meanwhile, reducing energy consumption (lower processor speed) can also increases profit. So, there is a trade off between higher speed and more energy saving
- RT-VM provisioning
  - After receiving a RT-VM request, system always select a node with minimum-price(highest speed, users pay according to execution time) of providing the RT-VM service. For same price, the node with less energy consumption will be provided

PM in Data Centers

- Power-aware Provisioning of Cloud Resources for Real-time Services
  - Power-aware VM provision schemes:
    - Lowest-DVS for VM Provisioning
      - Adjusts the processor speed to the lowest level at which RT-VMs meet their deadlines
      - Low acceptance rate, low power consumption
    - δ-Advanced-DVS for VM Provisioning
      - To overcome the low service acceptance rate of previous method, it operates the processor speed $\delta\%$ faster in order to increase the possibility of accepting coming RT-VM requests
    - Adaptive-DVS for VM Provisioning
      - Using M/M/1 model to predict the workload periodically
      - Adjust the processor speed adaptively

Performs Good in medium and light workload

Performs Good in a heavy workload

Future Work

- PM and resource allocation in the multi-core system
- Intelligent techniques to manage network resources efficiently
- Reduce he transition overhead caused by switching between different power states and VM migration
- Decentralized algorithm to provide scalability and fault tolerance
- PM in geographically distributed data center

Questions and comments

Leakage Current

- 1. Reverse-biased junction leakage current (IREV)
- 2. Gate induced drain leakage (IGIDL)
- 3. Gate direct-tunneling leakage (IG)
- 4. Subthreshold (weak inversion) leakage (ISUB)

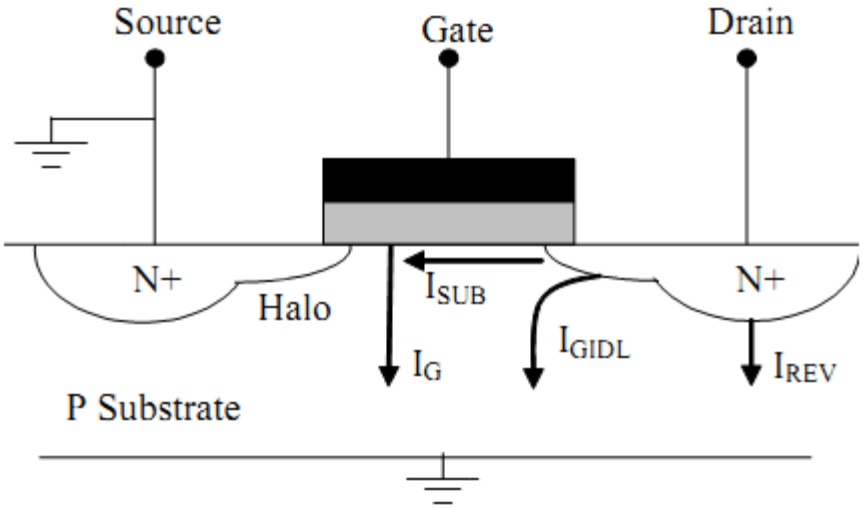
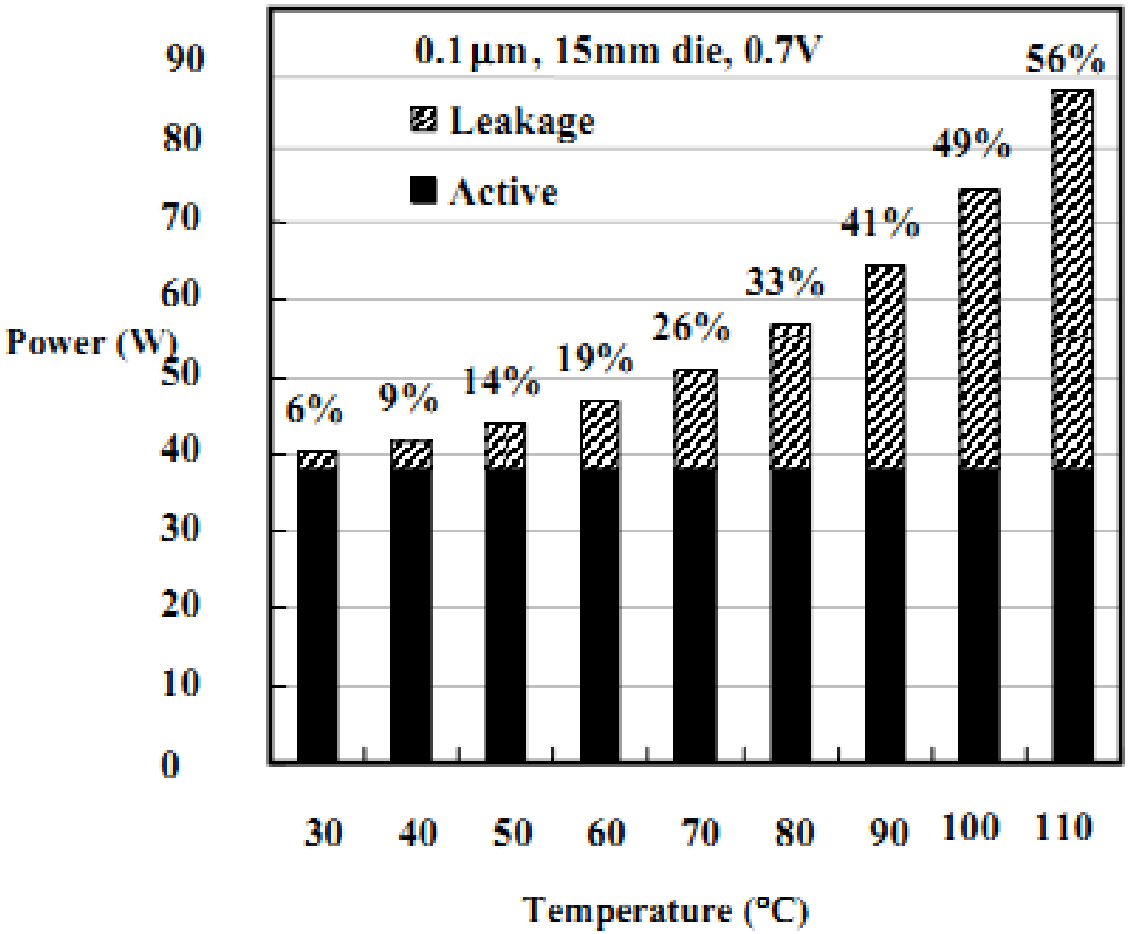


Figure 1: Leakage current components in an NMOS transistor.

Leakage Current

- Power consumption of a die as a function of temperature.



PM in Data Centers

- pMapper: Power and Migration Cost Aware Application Placement