On Satisfying Green SLAs in Distributed Clouds

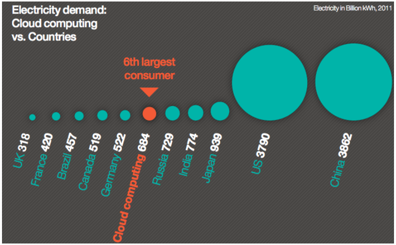
Ahmed Amokrane[1], M.Faten Zhani[2], Qi Zhang[2], Rami Langar[1], Raouf Boutaba[2], Guy Pujolle [1]

[1] LIP6, Pierre and Marie Curie University, Paris 6, France

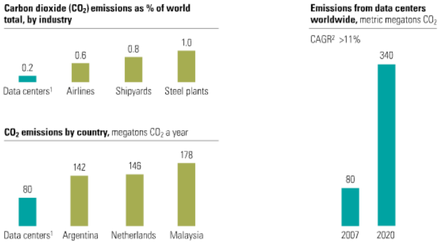
[2] University of Waterloo, Ontario, Canada

CNSM 2014, Rio de Janeiro, Brazil

Environmental Impact of Cloud

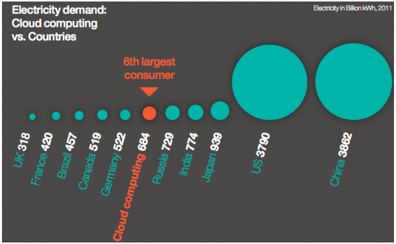


Source: Greenpeace

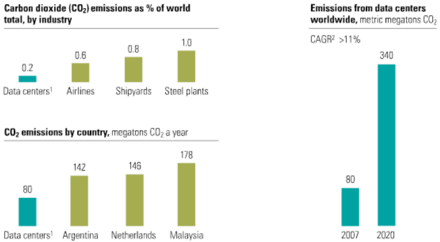


Source: Greenscroll.org

Environmental Impact of Cloud



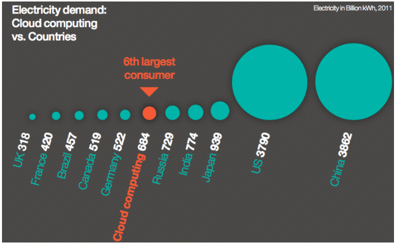
Source: Greenpeace



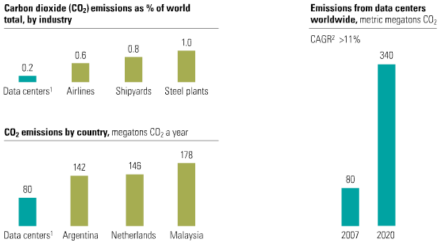
Source: Greenscroll.org

Energy Reduction = Cost Reduction

Environmental Impact of Cloud



Source: Greenpeace



Source: Greenscroll.org

Energy Reduction = Cost Reduction

Carbon Reduction = ?

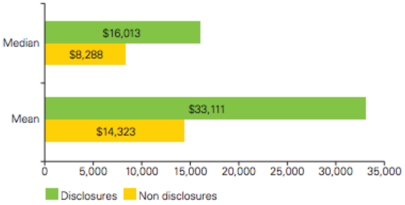
Why Do Carbon Emissions Matter for Companies?

- The amount of generated carbon impacts the company's value [KPMG]

Why Do Carbon Emissions Matter for Companies?

- The amount of generated carbon impacts the company's value [KPMG]
- Carbon disclosure has an impact on the company's value as well

Market Value (in $ millions)

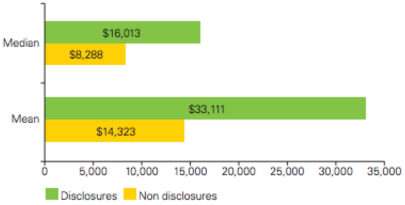


Source: CDP and S&P 500 reports

Why Do Carbon Emissions Matter for Companies?

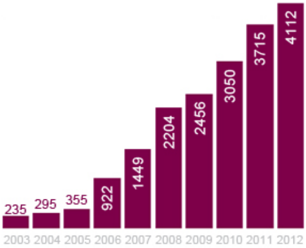
- The amount of generated carbon impacts the company's value [KPMG]
- Carbon disclosure has an impact on the company's value as well

Market Value (in $ millions)



Source: CDP and S&P 500 reports

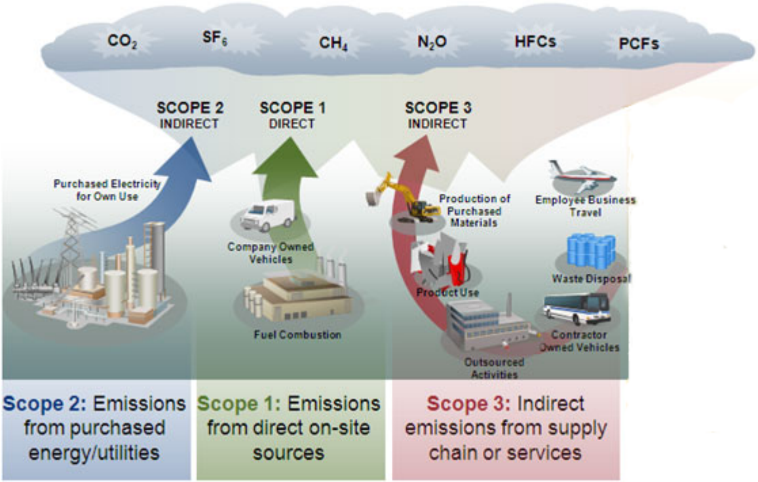
⇒



Source: Carbon Disclosure Project (CDP)

Carbon Emissions Scopes

Carbon emissions is everyone's responsibility, directly or indirectly



Explicit Specification of the Green Requirements

- In a Cloud environment
  - Cloud Providers (CPs): own and lease the infrastructure
  - Service Providers (SPs): use the infrastructure to offer services

Explicit Specification of the Green Requirements

- In a Cloud environment
  - Cloud Providers (CPs): own and lease the infrastructure
  - Service Providers (SPs): use the infrastructure to offer services

- A move towards explicit definition of the green targets by SPs
  - Limits in the carbon emission for the hosted services [1-5]
  - Minimum renewable energy to use [6]

[1] Cloud Selected Industry Group, Service Level Agreements expert subgroup, April 2013

[2] G. Laszewski et al. GreenIT Service Level Agreements,

[3] C. Bunse et al. GreenSLAs: Supporting Energy-Efficiency through Contracts,

[4] A. Galati et al. Designing an SLA Protocol with Renegotiation to Maximize Revenues for the CMAC Platform,

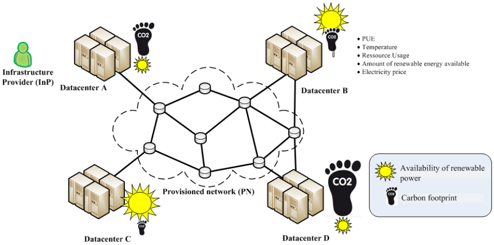
[5] C. Atkinson et al. Facilitating greener it through green specifications,

[6] M. Haque et al. Providing green SLAs in high performance computing clouds,

Outline

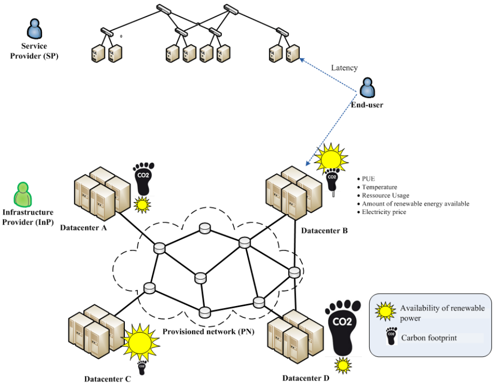
Outline

Problem Overview (SAVI)



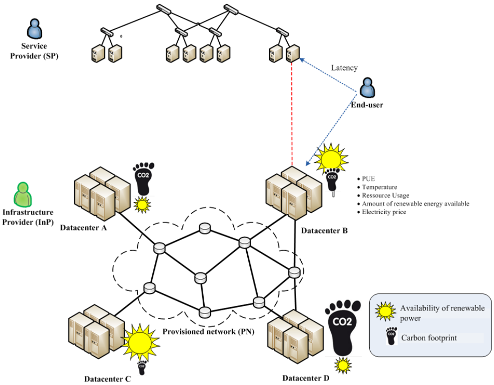
- Multiple data centers:
  - Different geographic locations
  - Different characteristics, e.g., Power Usage Effectiveness (PUE)
  - Difference in electricity price, carbon footprint for grid power, availability of onsite renewables

Problem Overview (SAVI)



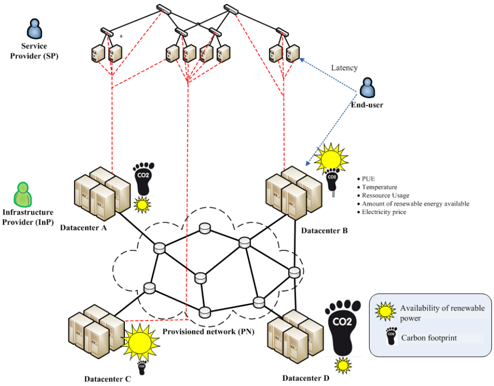
- Multiple data centers:
  - Different geographic locations
  - Different characteristics, e.g., Power Usage Effectiveness (PUE)
  - Difference in electricity price, carbon footprint for grid power, availability of onsite renewables
- VDC requests:
  - VMs (CPU, memory, disk), virtual links (bandwidth, delay)
  - **Green SLA terms**

Problem Overview (SAVI)



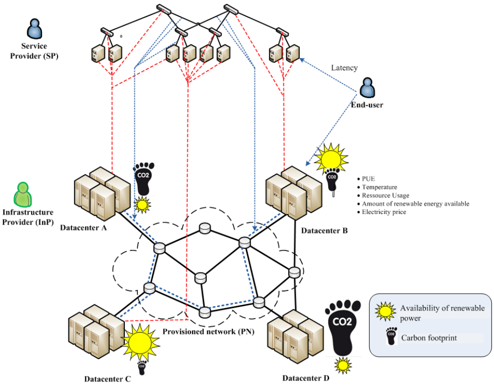
- Multiple data centers:
  - Different geographic locations
  - Different characteristics, e.g., Power Usage Effectiveness (PUE)
  - Difference in electricity price, carbon footprint for grid power, availability of onsite renewables
- VDC requests:
  - VMs (CPU, memory, disk), virtual links (bandwidth, delay)
  - **Green SLA terms**

Problem Overview (SAVI)



- Multiple data centers:
  - Different geographic locations
  - Different characteristics, e.g., Power Usage Effectiveness (PUE)
  - Difference in electricity price, carbon footprint for grid power, availability of onsite renewables
- VDC requests:
  - VMs (CPU, memory, disk), virtual links (bandwidth, delay)
  - **Green SLA terms**

Problem Overview (SAVI)



- Multiple data centers:
  - Different geographic locations
  - Different characteristics, e.g., Power Usage Effectiveness (PUE)
  - Difference in electricity price, carbon footprint for grid power, availability of onsite renewables
- VDC requests:
  - VMs (CPU, memory, disk), virtual links (bandwidth, delay)
  - **Green SLA terms**

Green SLA Specification

- Green SLA Specification: 2 components
  - SP specifies the **upper limit** of the carbon emission
    - The Carbon emissions due to the VDC should not exceed x tons for a period of time
  - Agreement on **the reporting period**
    - Periodically: week, month, every billing period (c.f. Open Data Center Alliance)
    - CPs should report to every SP the amount of carbon generated on his behalf

Green SLA Specification

- Green SLA Specification: 2 components
  - SP specifies the **upper limit** of the carbon emission
    - The Carbon emissions due to the VDC should not exceed x tons for a period of time
  - Agreement on **the reporting period**
    - Periodically: week, month, every billing period (c.f. Open Data Center Alliance)
    - CPs should report to every SP the amount of carbon generated on his behalf
- At the end of the reporting period, a CP reports:
  - The total Carbon Emissions
  - Carbon Emission per unit of resource
    e.g., tonsCO2/Gbps (Akamai), kgCO2/TB (Verizon), gCO2/search and gCO2/user per month (Google)
  - Carbon Emission per VDC

Outline

Context

- From the Cloud Provider standpoint
- Multiple data centers: Heterogenous data centers (PUE, renewables, electricity price)
- Dynamic arrival and departure of VDCs
- Green SLA terms of VDCs are defined for reporting periods of duration T^k
- A reporting period T^k is divided into time slots
    - Data centers characteristics fixed during a time slot

Mathematical Formulation

- Physical infrastructure: $G(V \cup W, E)$
  V: data centers, W: backbone network nodes, E: phy. links
- VDC Request j: $G^j(V^j, E^j)$
  V^j: VMs, E^j: virtual links
- A reporting period T^k is divided into time slots
- Decision variables:

$$x_{ik}^{j,t} = \begin{cases} 1 & \text{If the VM } k \text{ of the VDC } j \text{ is assigned} \\ & \text{to data center } i \text{ during time slot } t \\ 0 & \text{Otherwise.} \end{cases}$$

$$f_{e,e'}^{t} = \begin{cases} 1 & \text{If the backbone link } e \text{ is used to embed} \\ & \text{the virtual link } e' \text{ during time slot } t \\ 0 & \text{Otherwise.} \end{cases}$$

Mathematical Formulation

- For every reporting period T^k:

$$Maximize \quad \mathcal{R}_k - (\mathcal{D}_k + \mathcal{B}_k + \mathcal{M}_k + \mathcal{P}_k)$$

  - $\mathcal{R}_k$: the revenue during the reporting period T^k
  - $\mathcal{D}_k$: the embedding cost in data centers
  - $\mathcal{B}_k$: the embedding cost in the backbone network
  - $\mathcal{M}_k$: the migration cost
  - $\mathcal{P}_k$: the penalty cost

Subject to:

  - Capacity constraints in the physical infrastructure (data centers and backbone network)
  - Location constraints of some VMs
  - Migration restrictions: e.g., service disruption

Mathematical Formulation

- Optimization problem formulation (ILP)
  - Computation time-wise inefficient
  - Too frequently changing (every single time slot)

Outline

Overview

- At the time of request arrival:
  - VDC partitioning[1]:
    - Group the VMs with high bandwidth demand into the same partition
    - Minimize the bandwidth in the backbone network
    - Reuse the Louvain Algorithm
  - Partition Embedding based on the estimation of available renewables in different data centers
  - Admission Control: Reject requests whose profit is negative

[1] A.Amokrane, M.F. Zhani, R. Langar, R. Boutaba, G. Pujolle: "Greenhead: Virtual Data Center Embedding Across Distributed Infrastructures". Transactions on Cloud Computing, 2013

Overview

- At the time of request arrival:
  - VDC partitioning[1]:
    - Group the VMs with high bandwidth demand into the same partition
    - Minimize the bandwidth in the backbone network
    - Reuse the Louvain Algorithm
  - Partition Embedding based on the estimation of available renewables in different data centers
  - Admission Control: Reject requests whose profit is negative

- Periodic Reconfiguration: Follow the renewables

[1] A.Amokrane, M.F. Zhani, R. Langar, R. Boutaba, G. Pujolle: "Greenhead: Virtual Data Center Embedding Across Distributed Infrastructures". Transactions on Cloud Computing, 2013

Admission Control Algorithm

- Try embedding and detect violations
- Use worst case estimation of energy consumption of the VDC
- Infer an estimation of the carbon rate for the VDC
- Reject the request if the profit is negative

Algorithm 1 Admission Control Algorithm

```
IN: predictionWdW // the prediction window
IN: reconfigInterval // the reconfiguration interval
IN: vdc // the VDC to embed
wdw ← min(predictionWdw, reconfigInterval)
possible ← possibleToEmbed(vdc)
if possible then
    carbonRate ← getEstimationCarbonRate(wdw)
    carbonLimitRate ← vdc.carbonLimit/wdw
    if carbonRate ≤ carbonLimitRate then
        Accept vdc
    else
        //Verify if profit can be made
        estimatedCost ← estimatePowerCost(vdc)
        if revenue(vdc) × (1 − refundFactor) − estimatedCost > 0 then
            Accept vdc
        else
            Reject vdc
        end if
    end if
else
    Reject vdc
end if
```

Partition Embedding Algorithm

- Greedy embedding of partitions
- Dijkstra's algorithm with backbone cost as a metric

Algorithm 2 Greedy VDC Partitions Embedding Across Data Centers

IN: $G(V \cup W, E)$, $G^j_M(V^j_M, E^j_M)$
for all $i \in V$ **do**
    $ToDC[i] \leftarrow \{\}$
end for
for all $v \in V^j_M$ **do**
    $S_v \leftarrow \{i \in V$ / i satisfies the location constraint$\}$
end for
for all $v \in V^j_M$ **do**
    $i \leftarrow s \in S_v$ with the smallest cost $getCost(s, v)$, and $LinksEmbedPossible(s, v) = true$
    if no data center is found **then**
        return $FAIL$
    end if
    $ToDC[i] \leftarrow ToDC[i] \cup \{v\}$
    for all $k \in N(v)$ **do**
        if $k \in ToDC[i]$ **then**
            $ToDC[i] \leftarrow ToDC[i] \cup \{e_{vk}\}$
        else
            if $\exists l \neq i \in V$ / $k \in ToDC[l]$ **then**
                Embed e_{vk} in G using the shortest path
            end if
        end if
    end for
end for
return $ToDC$

Dynamic Reconfiguration Algorithm

- Idea: Offload VMs to data centers where renewables are available
- Hints:
  - Use the actual power consumption in data centers
  - Estimate the available renewables (prediction window of around 4 hours)
  - Classify the partitions in increasing order of migration costs
  - Migrate in the increasing order of costs
  - Destination choice: reduce the bandwidth in the backbone network

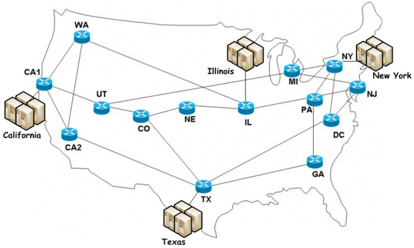
Algorithm 3 Greedy Partition Migration Across Data Centers

```
IN: predictionWdW // the prediction window
IN: reconfigInterval // the reconfiguration interval
wdw ← min(predictionWdW, reconfigInterval)
for all i ∈ V do
    Diff[i] ← EstimateRenewables(wdw, i) − FutureConsumption(wdw, i)
    if Diff[i] < 0 then
        part[i] ← list of partitions in i sorted by migration cost
    end if
end for
for all i ∈ V, Diff[i] < 0 do
    while ∃ k ∈ V, Diff[k] > 0 do
        p ← part[i].first
        D ← {k ∈ V, Diff[k] > 0}
        done ← false
        while !done && D ≠ φ do
            //Take the data center with the minimum cost in the backbone network after migration
            dest ← minBackboneCost(D)
            Migrate(p, dest)
            if successful migration then
                done ← true
                Update Diff[dest] and Diff[i]
            else
                D ← D\{dest}
            end if
        end while
    end while
end for
```

Outline

Simulation Settings: Parameters

- Physical Topology
  - 4 Data centers connected through NSFNet
  - Traces for renewable power and carbon emissions



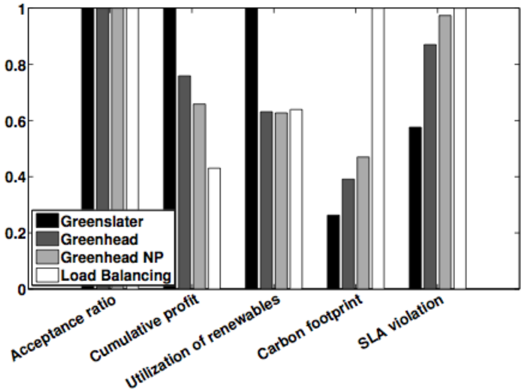
- VDC Requests
  - Poisson Arrivals, Exponential lifetime (24 hours)
  - $U(10, 50)$ VMs, $U(1, 4)$ cores, virtual links $U(10, 50Mbps)$
  - Carbon constraints $U(5, 20kg)$ CO_2 / 24 hours
- SLA violation: the CP pays back 50% of the bill

Simulation Settings: Baselines

- Greenhead:
  - Same partitioning, Greedy VDC embedding
  - Objective: Reduce operational costs and carbon emissions
  - No dynamic reconfiguration, No explicit Green SLA consideration
- Greenhead No Partitioning (NP):
  - Greedy VDC embedding
  - Objective: Reduce operational costs and carbon emissions
  - No Partitioning, No dynamic reconfiguration, No explicit Green SLA consideration
- Load Balancing:
  - Load balance the VDCs across the data centers
  - No dynamic reconfiguration, No Green SLA consideration

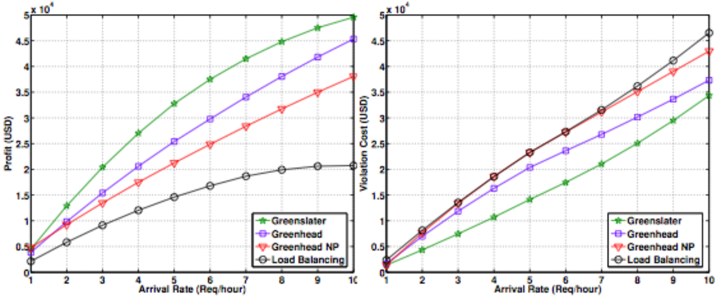
Results on Different Metrics

- Higher profit and higher utilization of renewables
- Reduced SLA violations and reduced carbon footprint



Impact of the Arrival Rate

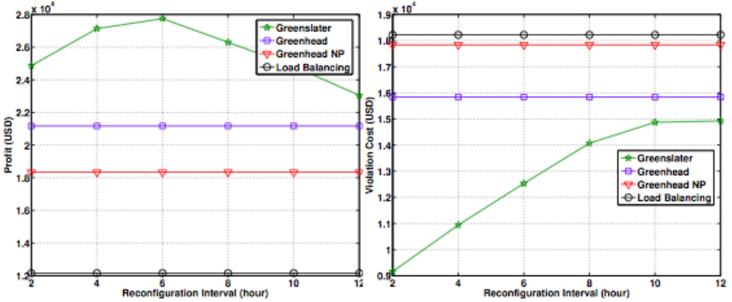
- Higher profit and reduced SLA violation costs



(a) Cumulative profit (b) SLA violation cost

Impact of the Reconfiguration Interval

- Higher profit and reduced SLA violation costs
- Optimal reconfiguration interval around 6 hours



(a) Cumulative profit (b) SLA violation cost

Outline

Conclusion

- Green SLAs in clouds
- Greenslater:
  - Partitioning, Admission Control, Embedding
  - Dynamic reconfiguration to follow the renewables
  - Satisfying Green SLAs
  - Higher profit
- Future work:
  - Consider proportional violations costs
  - Study the pricing model Green SLA vs. traditional SLA
  - Consider re-negotiation instead of rejecting requests