

263-3855-00: Cloud Computing Architecture

Section 12: Sustainability in the Cloud

Swiss Federal Institute of Technology Zurich (ETH Zürich)

Last Edit Date: 05/31/2025

Disclaimer and Term of Use:

1. We do not guarantee the accuracy and completeness of the summary content. Some of the course material may not be included, and some of the content in the summary may not be correct. You should use this file properly and legally. We are not responsible for any results from using this file.
2. This personal note is adapted (derived) from the slides and codes from *Professor Ana Klimovic and people from ETH Zurich Systems Group*. Please [contact us](#) to delete this file if you think your rights have been violated.
3. This work is licensed under a [Creative Commons Attribution 4.0 International License](#).

I. The Current Situation with Cloud Infrastructure

1. Industrial Scale of Computing

- Computing has reached industrial scales, comparable to building airports, highways, and metro systems.
- A different approach to thinking about computing is needed by the public, decision-makers, and computer scientists.
- The common perception does not align with this reality.

2. Rising IT Energy Consumption

- The CEO of the UK's National Grid warned of datacenters' increasing power demand, predicting a 500% increase in consumption over a decade.
- Amazon acquired Cumulus Data's atomic datacenters for \$650 million, securing 480 MW of power from the Susquehanna plant.

3. AI as a Major Contributor to Energy Consumption

- The training cost and compute for AI models have significantly increased, with models like Google's Gemini Ultra and OpenAI's GPT-4 showing high training costs and compute requirements.
- Microsoft plans to increase its capital expenditure by over 50% year-on-year to \$50 billion, with talks of spending \$100 billion on an AI supercomputer.
- Alphabet's capital expenditure in the first quarter was \$12 billion, primarily driven by investment in technical infrastructure for AI.

4. The Paradox of AI and Green Transition

- While AI is seen as a tool to accelerate the energy transition by managing complexity and improving system efficiency, its own energy consumption is a concern.
- Microsoft's environmental sustainability report indicated that while Scope 1 and 2 emissions decreased by 6.3% from their 2020 baseline, indirect emissions (Scope 3) increased by 30.9%.
- In aggregate, Microsoft's total emissions (Scopes 1-3) are up 29.1% from the 2020 baseline.
- This increase in emissions is attributed to significant investments in AI, with Microsoft planning to invest \$80 billion on AI-enabled data centers in fiscal 2025.

5. Cloud Design Philosophy and Environmental Impact

- The cloud is primarily built for performance and economic efficiency.
- Environmental impact is often treated as a cost issue rather than a social concern.

- Cloud infrastructure is based on legacy architectures and software, which can run counter to environmental goals.
- For data security, data-bearing devices are commonly physically destroyed, even with advanced encryption, preventing reuse and recovery of components like rare earth magnets.

II. Why IT is Not Efficient

1. Contrast with Classical Engineering

- Classical engineering designs physical objects, whereas computer science designs virtual objects.

2. The "CS Way of Calculating"

- An analogy is used to illustrate inefficiency: transporting 200 Kg of cargo. A truck needs one trip, while an F1 car might need three trips (carrying 100 Kg, returning, then carrying the other 100 Kg).
- If the F1 car is significantly faster, it might complete the task in the same time, leading to the misleading conclusion that the F1 car is "better off" despite being less efficient.

3. Common Practices in IT

- It is common practice to distribute workloads across clusters, often without sufficient justification for small data sizes.
- The belief that "nobody ever got fired for using Hadoop on a cluster" highlights the tendency to default to distributed systems.
- Many algorithms are complex to scale out and expensive in terms of human engineering.

4. The "COST" Metric

- "COST" stands for "Configuration that Outperforms a Single Thread."
- This metric measures the hardware configuration required for a platform to outperform a competent single-threaded implementation.
- It weighs scalability against system overheads to show actual performance gains, not just parallelizable overheads.
- Studies have shown that unoptimized distributed implementations can "scale" better due to poor single-node performance, illustrating that scalability doesn't always equate to efficiency.

5. Examples in Graph Processing

- For 20 PageRank iterations on the "twitter_rv" dataset, a single-thread (smarter) implementation takes 110s, while Spark on 128 cores takes 857s, Giraph 596s, GraphLab 249s, and GraphX 419s.

- For label propagation to fixed-point on the "twitter_rv" dataset, a single-thread (smarter) implementation takes 15-30s, while Spark on 128 cores takes 1784s, Giraph 200s, GraphLab 242s, and GraphX 251s.

6. Memory Inefficiencies

- DRAM can account for a significant portion of server costs (50% in Azure, 40% at Meta).
- Up to 25% of DRAM can become stranded as more cores are allocated to VMs.

7. The Price of Distribution

- Profiling a warehouse-scale computer reveals significant cycles spent on remote procedure calls (RPCs), protobuf, and allocation, highlighting the overhead of distributed systems.

8. Externality of Pollution

- Externality refers to costs incurred by a third party due to the activities of others, such as pollution not paid by the producer or consumer but by everybody.
- Externalities and real costs cannot be ignored, as exemplified by TPC benchmarks measuring performance in transactions per time per dollar.

III. Green IT

1. IT as an Industry

- The IT industry needs to adopt a large, industrial-scale approach to green computing.
- Similar to changes required for cars, planes, and energy production, the IT industry must undergo fundamental changes to reduce resource use.
- This change is likely to be resisted by the IT industry.

2. Types of Emissions (Scopes)

- Scope 1: Direct emissions from economic activity (e.g., house heating).
- Scope 2: Emissions from purchased energy (e.g., consumed electricity).
- Scope 3: Indirect emissions related to the entire activity (e.g., embodied carbon of equipment).

3. Embodied Carbon

- Hardware lifecycle (manufacturing, transport, use, recycling) accounts for over 98% of Apple's total emissions.
- Manufacturing alone accounts for 74% of total emissions, while hardware use is 19%.
- Carbon output from manufacturing integrated circuits (SoCs, DRAM, NAND flash memory) is higher than from hardware use.
- TSMC (chip manufacturer) uses almost 5% of all Taiwan's electricity (projected to rise to 7.2% in 2022) and 63 million tons of water in 2019.

- Intel's Ocotillo campus in Arizona produced nearly 15,000 tons of waste (60% hazardous), consumed 927 million gallons of fresh water, and used 561 million kilowatt-hours of energy in the first three months of a year.

4. Microsoft's Scope 3 Emissions

- Microsoft's 2024 Environmental Sustainability Report states that Scope 1 and 2 emissions decreased by 6.3% from their 2020 baseline.
- However, indirect emissions (Scope 3) increased by 30.9%.
- In aggregate, Microsoft's total emissions (Scopes 1-3) are up 29.1% from the 2020 baseline.

IV. Cloud Architecture is Inefficient

1. Evolution from Traditional to Cloud Architecture

- Initial architectures often had local compute and storage.
- This evolved to separation of compute and storage.
- Additional functionalities like user profiles, sponsored content, and ads were added, leading to further architectural layers.
- Load balancers and caching layers were introduced to compensate for design problems and improve performance.

2. Designing Around Problems

- The prevalence of MapReduce-like abstractions on commodity clusters is noted.
- The phrase "nobody ever got fired for using Hadoop on a cluster" signifies a common, perhaps uncritical, adoption of distributed systems.
- Many analytics jobs, particularly iterative machine learning algorithms, are complex to scale out and require significant human engineering.

V. Moving Forward

1. Imbalances in Modern Computers

- Modern computers exhibit significant imbalances, including the memory wall, inadequate CPUs for specialized tasks, legacy buses and interconnects, large-scale distribution overheads, and performance differences between computing devices.
- These problems are often "solved" by adding more machines.

2. Takeaway Message

- "There is no performance problem that cannot be solved by adding more machines, there is no efficiency problem that cannot be solved by removing machines."

3. From Scale-Out to Scale-Up

- Scale-out has been the dominant cloud model for decades.
- It may be time to consider scale-up by building larger machines for large jobs, running on larger machines/VMs, and avoiding RPCs, data center tax, complexity, and data exchanges.
- Hardware advancements enable this option.
- The question arises as to how many workloads genuinely require distribution, and whether "good enough" is a viable approach.
- Examples of scale-up include Project Ceiba and NVIDIA Blackwell GPUs for AI supercomputers.

4. Disaggregation

- Disaggregated server architectures allow for independent refresh of CPU/DRAM modules and NIC/drives modules.
- This enables replacing only specific modules instead of the entire unit.
- CXL (Compute Express Link) can enable the reuse of DRAM, which has a longer lifespan than machines.
- Intel's vision for disaggregation includes evolving from physical aggregation to fabric integration and eventually subsystem aggregation (storage, compute, memory).
- This approach aims for platform flexibility, higher density, and higher utilization.

5. Accelerators and Architectural Changes

- Hardware is rapidly evolving with new CPU architectures, ISAs, interconnects, and specialized accelerators (e.g., Microsoft Azure Maia 100, Cloud TPU v5p, AWS Graviton CPUs).
- These advancements can potentially increase the efficiency of data processing.
- Architectural changes, such as support for multiple accumulator and output types and varying precision, are being explored.

6. Function Shipping vs. Data Shipping

- The concept of "function shipping" (e.g., SQL In Silicon, Amazon AQUA) is presented as an alternative to "data shipping."
- This approach aims to bring computation closer to the data, minimizing data movement and improving efficiency.