

The Case for Time-Shared Computing Resources

Pierre Jacquet,¹ Adrien Luxey-Bitri²

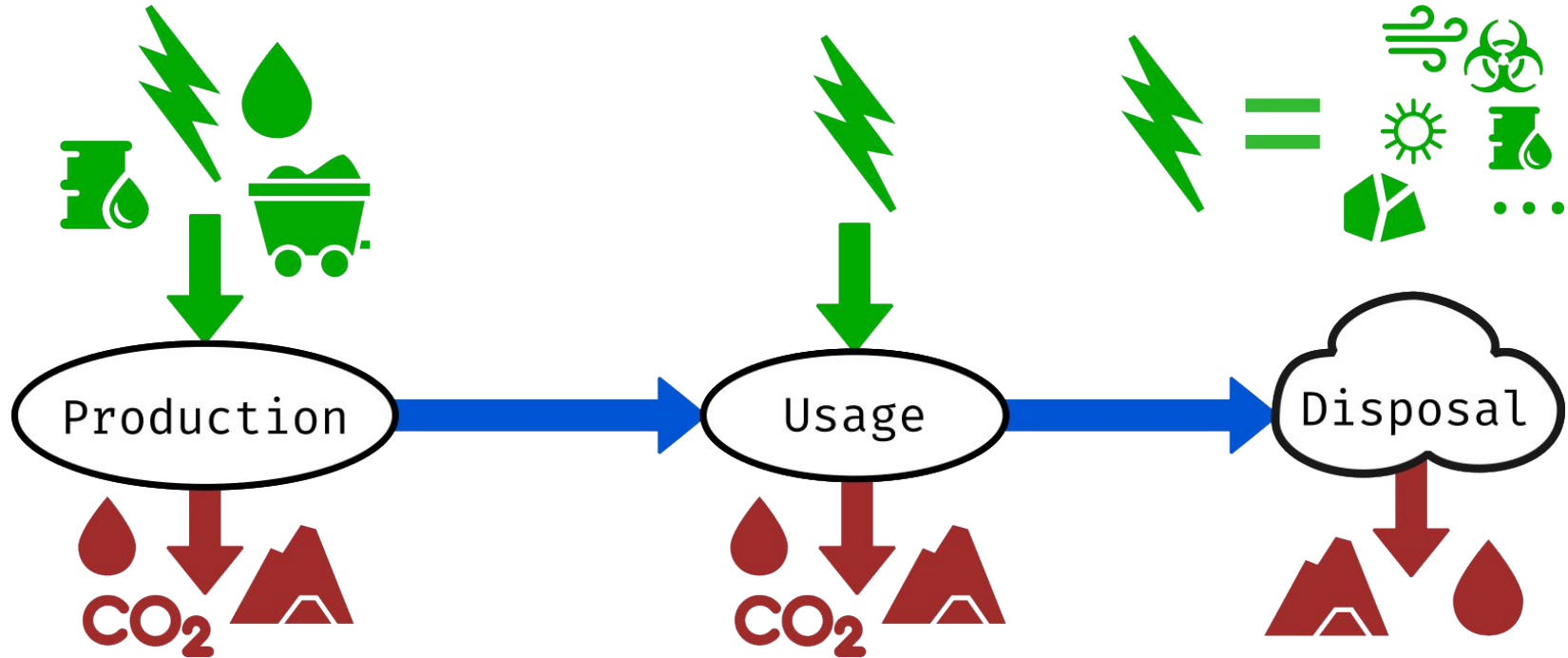
ÉTS Montréal, Université du Québec (Canada)¹,
Université de Lille, Inria (France) - Deuxfleurs association²



ÉCOLE DE
TECHNOLOGIE
SUPÉRIEURE
Université du Québec



Introduction



ICT cannot afford its current material footprint forever

Introduction

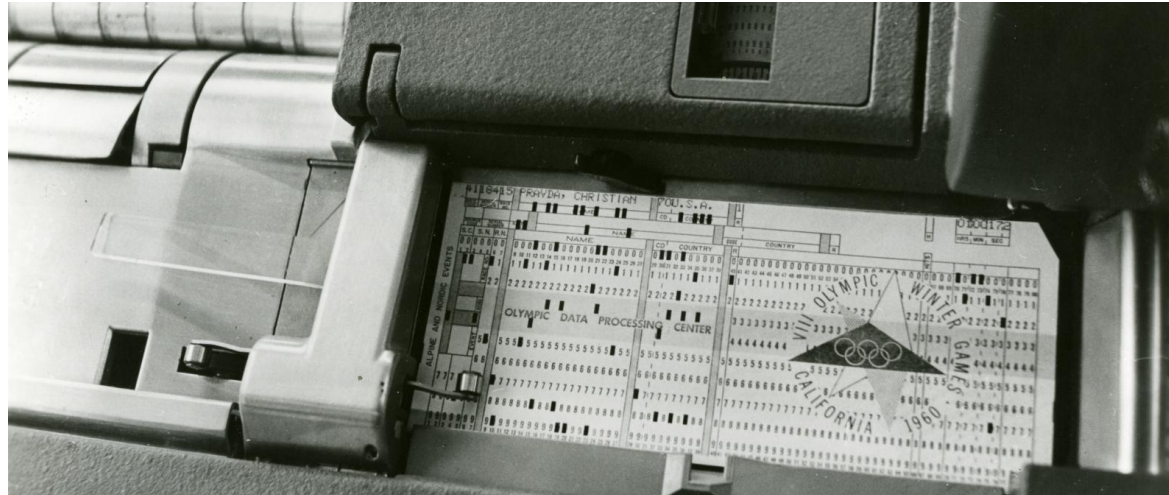


Hosting facilities will soon have to **fit into a defined material envelope**

Brief History of Resource-Sharing in ICT

Sharing is as old as first programmable systems

- The same machine can be used **for different purposes**

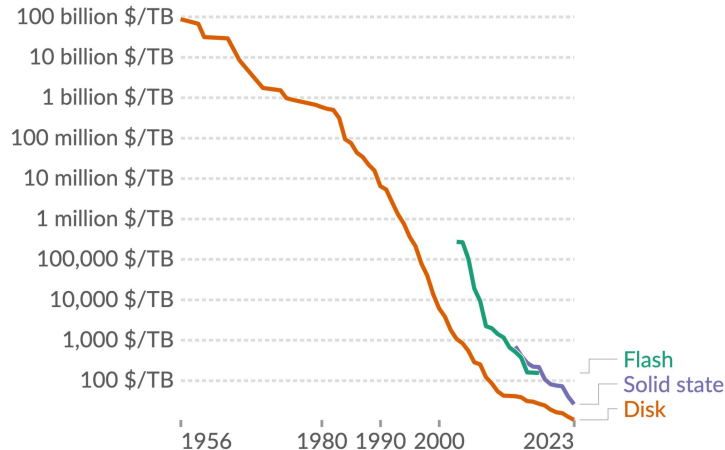


IBM Mainframe

Brief History of Resource-Sharing in ICT

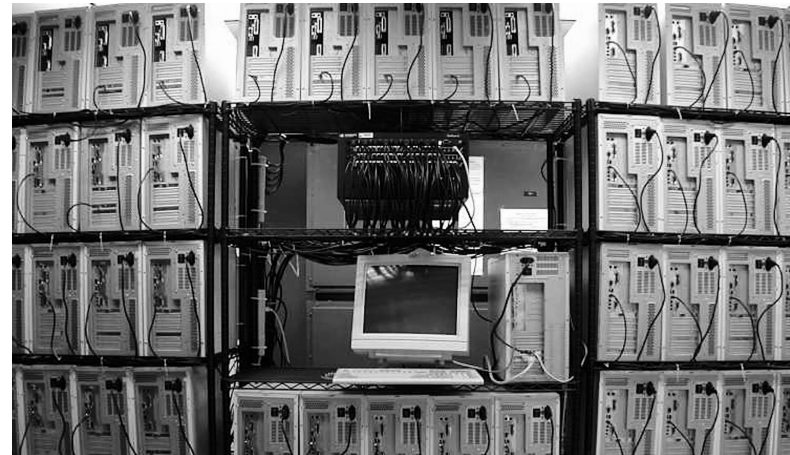
The distributed era leads to greater resource dedication

- Reasons include: Performance consideration, Security consideration, Cost Decrease, etc.



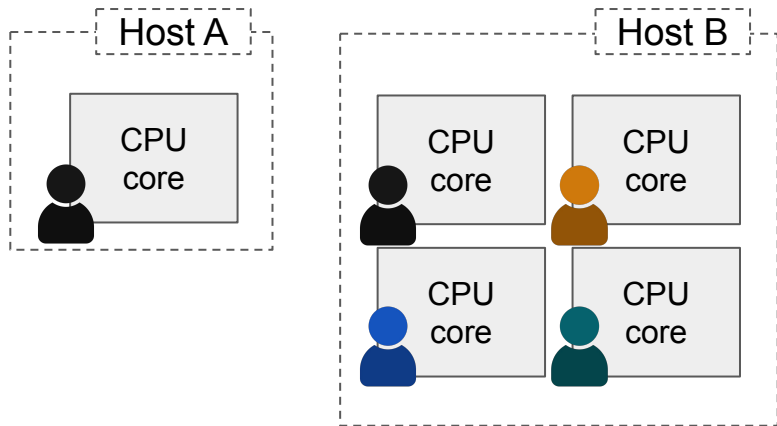
Data source: John C. McCallum (2023)

CC BY



Defining Principles of Time-Shared Resources

What should we share?
(i.e. unit to consider)



Centralising is not necessarily sharing

How should we share it?
(i.e. policies to consider)

Sequential Sharing



Concurrent Sharing



vCPUs vs. functional units



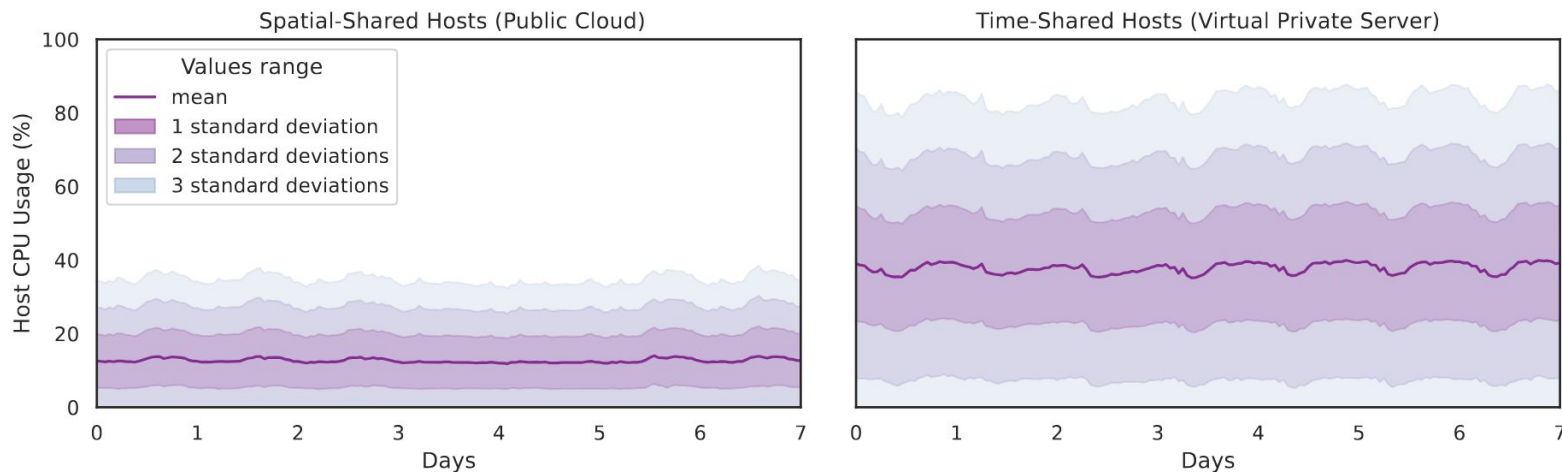
Share an *envelope* or a *software*

Applications in Different Contexts

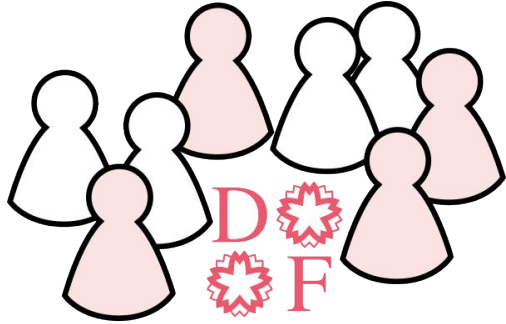


Applications in Different Contexts

- For a Cloud Provider, shared products are more efficient



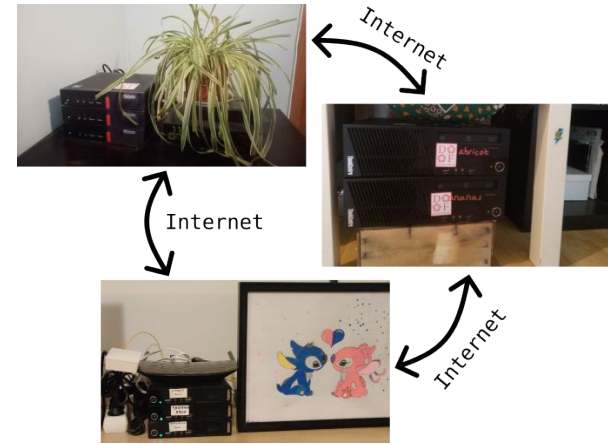
Applications in Different Contexts



The community
decides upon ...



The services
running on ...



**The shared
infrastructure**

Operational limits (maintenance time, democratic decision-making...)

➡ **adequate services** (static sites, tight storage quotas...)

➡ **simple infrastructure** (few old machines, standards...)

Take-aways

- Soon back into **constrained & shared computing** models?
- **Time-sharing** as a means to fit services into a **fixed material envelope**?
- **Grassroots ICT** can emancipate & fit within **planetary boundaries**
- **Infrastructure** ↔ **Superstructure**

