Enabling Sustainable Clouds: The Case for Virtualizing Energy System

Noman Bashir - Postdoctoral Research Fellow

University of Massachusetts Amherst

Talk slides for Low Carbon and Sustainable Computing (LOCOS) seminar at U of Glasgow

November 3, 2022

Research Overview

Sustainable Computing

Sustainable Computing - Without the Hot Air: HotCarbon'22.

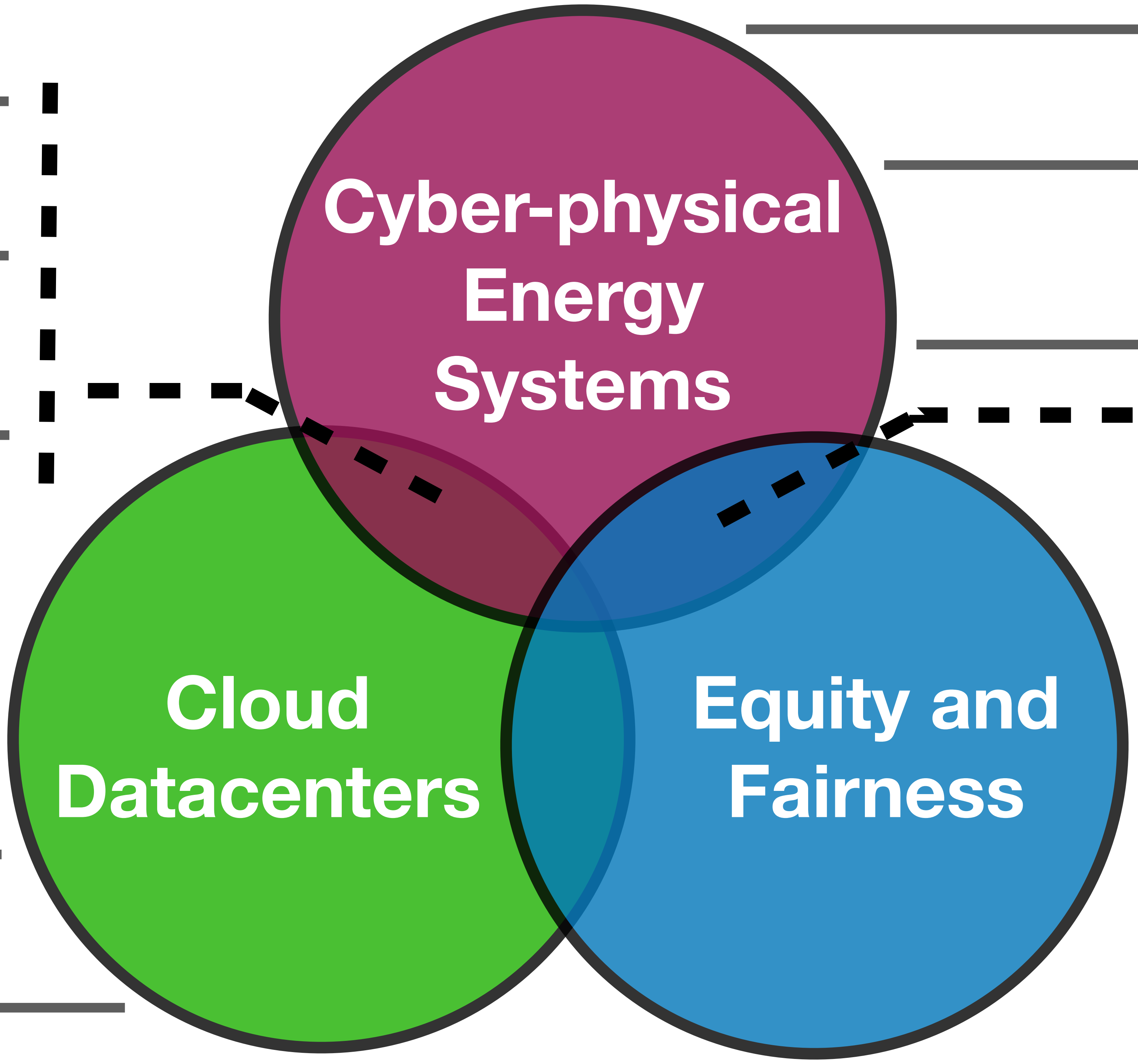
Ecovisor: A Virtual Energy System for Carbon-efficient Applications : SoCC'21, ASPLOS'23.

CarbonScaler, Acclimator, Delen: Under-review at EuroSys'23, SIGMETRICS'23, IoTDI'23.

Resource Overcommitment in Google's Datacenters: EuroSys'21.

Cost and Wait-time Optimization for Hybrid Cloud Datacenters: SC'20, IC2E'20, SoCC'21.

Non-intrusive Power Monitoring in Cloud Datacenters: Under-review at SIGMETRICS'23.



Solar PV Performance Modeling and Forecasting: BuildSys'20, MASS'20.

Programmable Networked Energy Systems: BuildSys'17, BuildSys'18, IGSC'20, COMPASS'20.

Electrification in Sub-Saharan Africa: COMPASS'20, e-Energy'19.

Equity-aware Energy Transition

Fair Control of Distributed Solar Capacity: BuildSys'17.

Residential Heating Decarbonization: BuildSys'22.

Network- & Equity-aware Gas Network Decarbonization: Under-review at Energy Informatics Review and e-Energy'23.

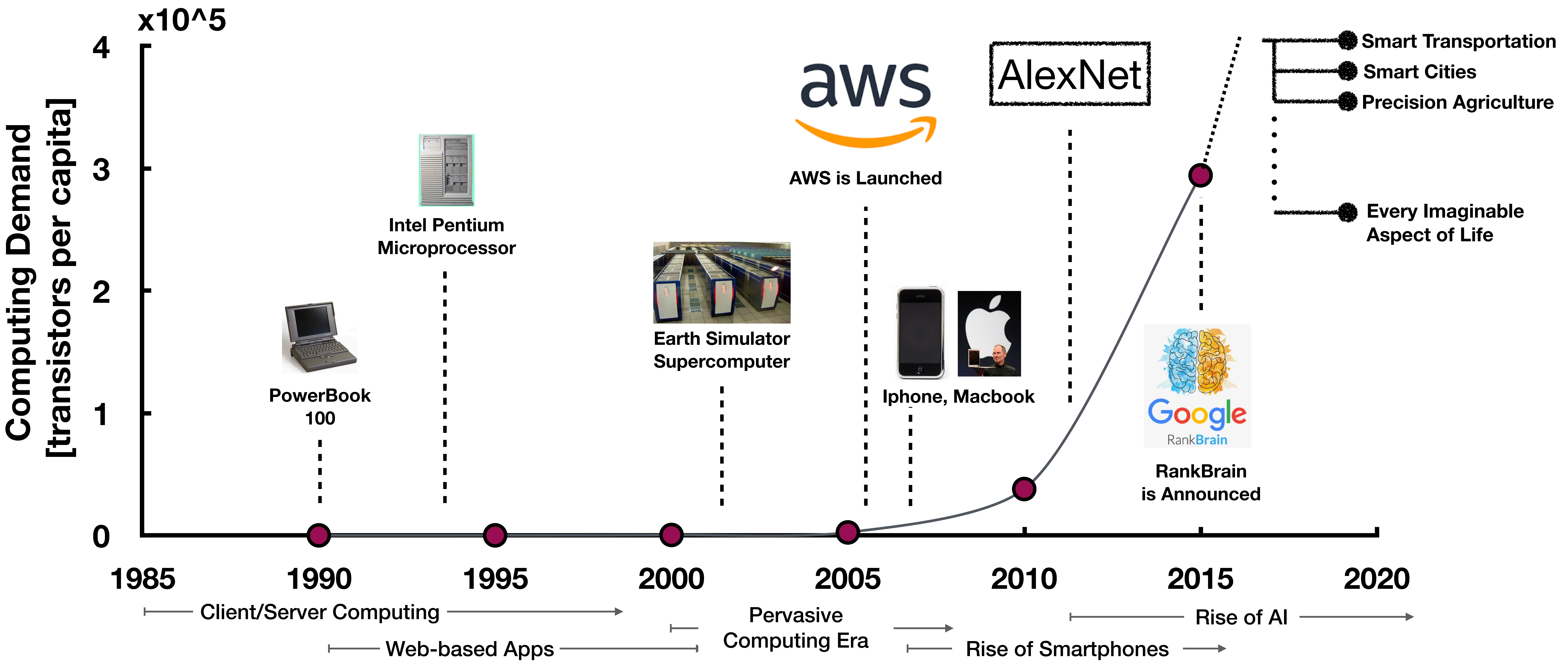
Sustainable Computing - Without the Hot Air*

*** Title inspired by the book Sustainable Energy - Without the Hot Air**

Noman Bashir - University of Massachusetts Amherst
Appeared at HotCarbon’22

Computing's Demand is Growing Exponentially

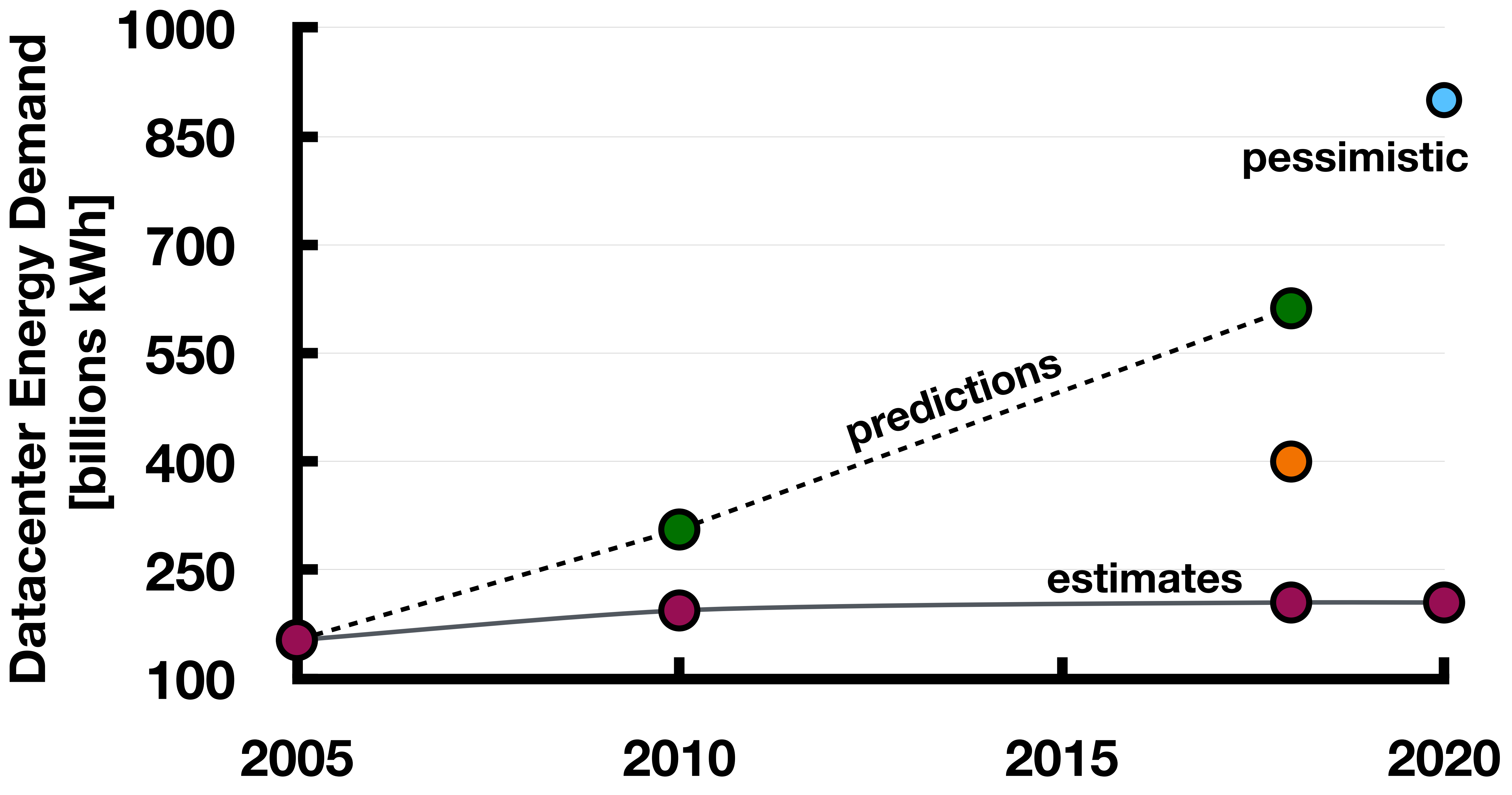
- Society continues to find useful applications



Source: "Unimaginable Output: Global Production of Transistors" - Darrin Qualman

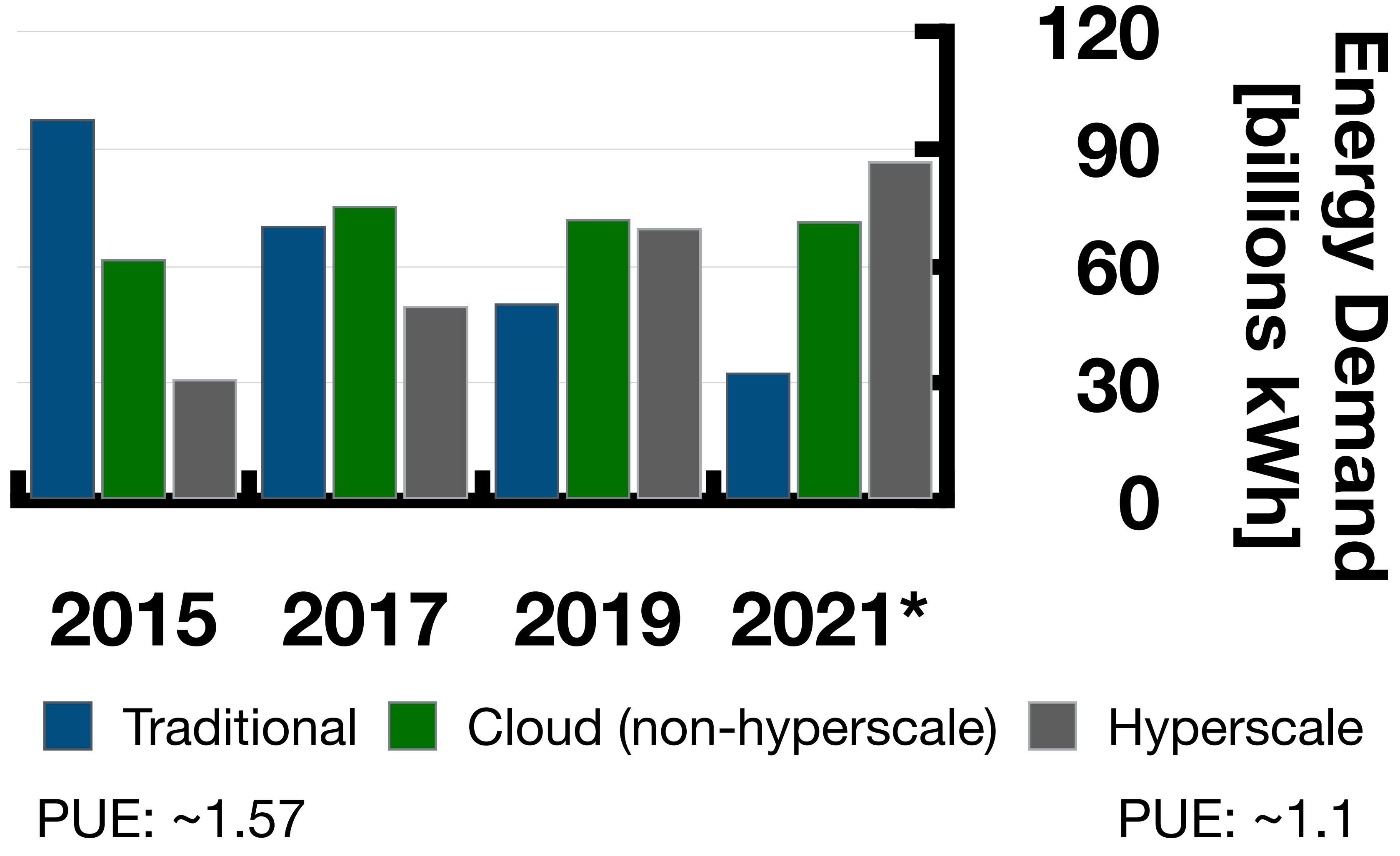
Exp. Demand ⇏ Exp. Energy Consumption

- Most optimistic estimates suggest 6% increase from 2010-2018



EPA Report to Congress on Server and Data Center Energy Efficiency (2007)
Recalibrating Global Data Center Energy-use Estimates - Eric Masanet (2020)
Efficiency Gains are Not Enough: Data Center Energy Consumption Continues to Rise Significantly - Ralph Hintemann (2018)

Shift from Traditional Datacenters to Cloud

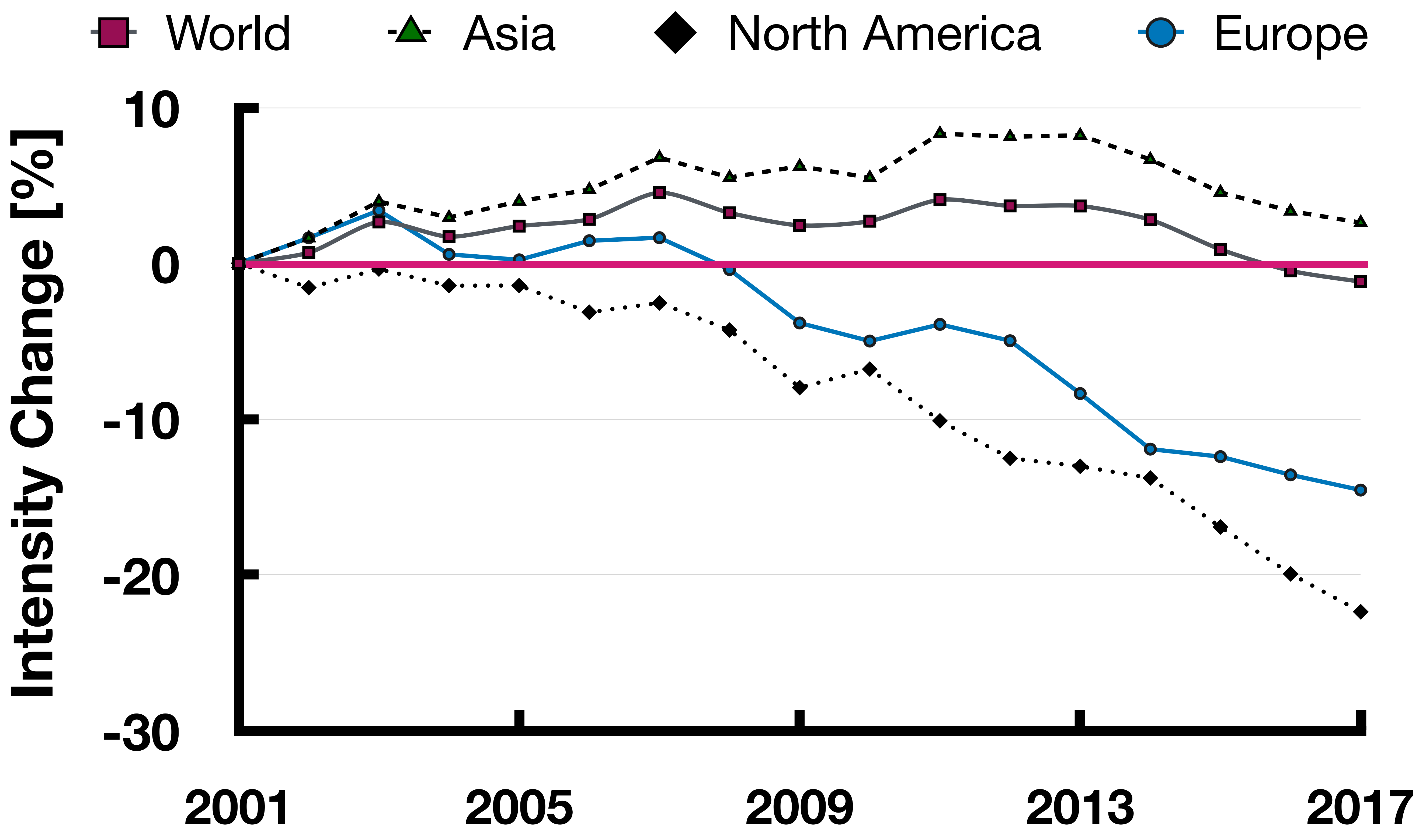


Demand Accelerating
vs
Energy-efficiency Gains Slowing Down

Source: Global data centre energy demand by data centre type, 2015-2021 - IEA

Grid's Carbon Intensity Has Been Decreasing

- Energy's carbon efficiency in the US has improved by 45.6% over 2001-2017



$$\text{Computing's Carbon Efficiency} = \frac{\text{Computing's Energy Demand}}{\text{Energy's Carbon Efficiency}}$$

2.33% increase in Carbon-efficiency
0.65% increase in Energy Demand
-> 1.64% decrease in Carbon Footprint

- **What if optimistic estimates are incorrect?**
- **How about the rest of the world?**

Source: Ember Global Electricity Review (2022)
Source: BP Statistical Review of World Energy
Source: Ember European Electricity Review (2022)

Algorithmic Efficiency can be further improved, but has limits

Industry has strong incentive to improve the **algorithmic efficiency**

Recent focus on ML training and Crypto-mining

[bounded]

[unbounded]

Datacenter capacity **increased by 6X** from 2010-2018

Crypto-mining and ML demand is **outpacing Moore's law**

Industry has strong incentive to **maintain and accelerate growth**

$$\textbf{Carbon Footprint} = \frac{\textbf{Cycles per Unit Work} \times \textbf{Total Units of Work}}{\textbf{Computing's Energy Efficiency} \times \textbf{Energy's Carbon Efficiency}}$$

[Koomey's Law: Energy efficiency doubles every 1.5-2.6 years**]** transition to cloud, dedicated hardware

[Laundar's Principle: Theoretical limit to be reached in 2050, practical sooner**]**

[Jevon's Paradox: Historically, gains in efficiency have not reduced demand**]**

[bounded]

[unbounded]

Zero-carbon energy means **carbon efficiency can be infinite**

Industry has helped subsidize zero-carbon energy

Carbon Accounting and Attribution Methods

- **Carbon offsets:** offset carbon-intensive grid energy with the use of zero-carbon energy at another ***location*** and ***time***

All Carbon Offset Subsidize Renewable Energy,
But Stricter the Carbon Offset, the Better

Andrew Chien's Good → Better → Best

24/7 carbon offsets

- **Google aims to be "Carbon Free" by 2030**
- **Piloted "Time-based Energy Attribute Certificates (TEACs)"**
- **Matching on the same grid in the same hour of the day**

Loosest → **Stricter** → **Strictest**

Carbon-neutral since inception

Annual, location-agnostic

- **Subsidize the adoption of renewable energy across the world**
- **Still cause significant amount of direct carbon emissions**

- **24/7 matching is not the panacea**
- **Carbon emissions should be attributed to all loads based on their energy consumption**
- **You are carbon free when society is carbon free**

Zero Carbon Grid

Accounting for and Reducing Embodied Carbon

- Carbon emissions from producing products or services, e.g., buildings facilities, manufacturing servers

Embodied	**Operational**
• Your embodied is someone else's operation • Incentivizes buying less or buying different	• Operational is completely under your control • Operational emissions are not a solved problem

- Embodied and operational emissions are **NOT additive**
- One is **NOT** more **important** than the **other**
- Focus on embodied can **distract** from operational

Implications for Sustainable Computing

Enable Visibility into Carbon

- WattTime and electricityMap
- Average vs marginal carbon?

Leverage Computing's Flexibility

- Change how we operate
- Use computing to balance grid

Pre-requisite

Carbon Intensive Grid

Carbon Free Grid

Clarify Misunderstandings

- "Carbon-free", "carbon-neutral", "zero-carbon", "100% renewable" mean different things
- Confusing terminology gives false impressions

Shift "Focus" from Energy to Carbon

- No direct nor financial incentive
- Indirect incentive exist

Key Takeaways

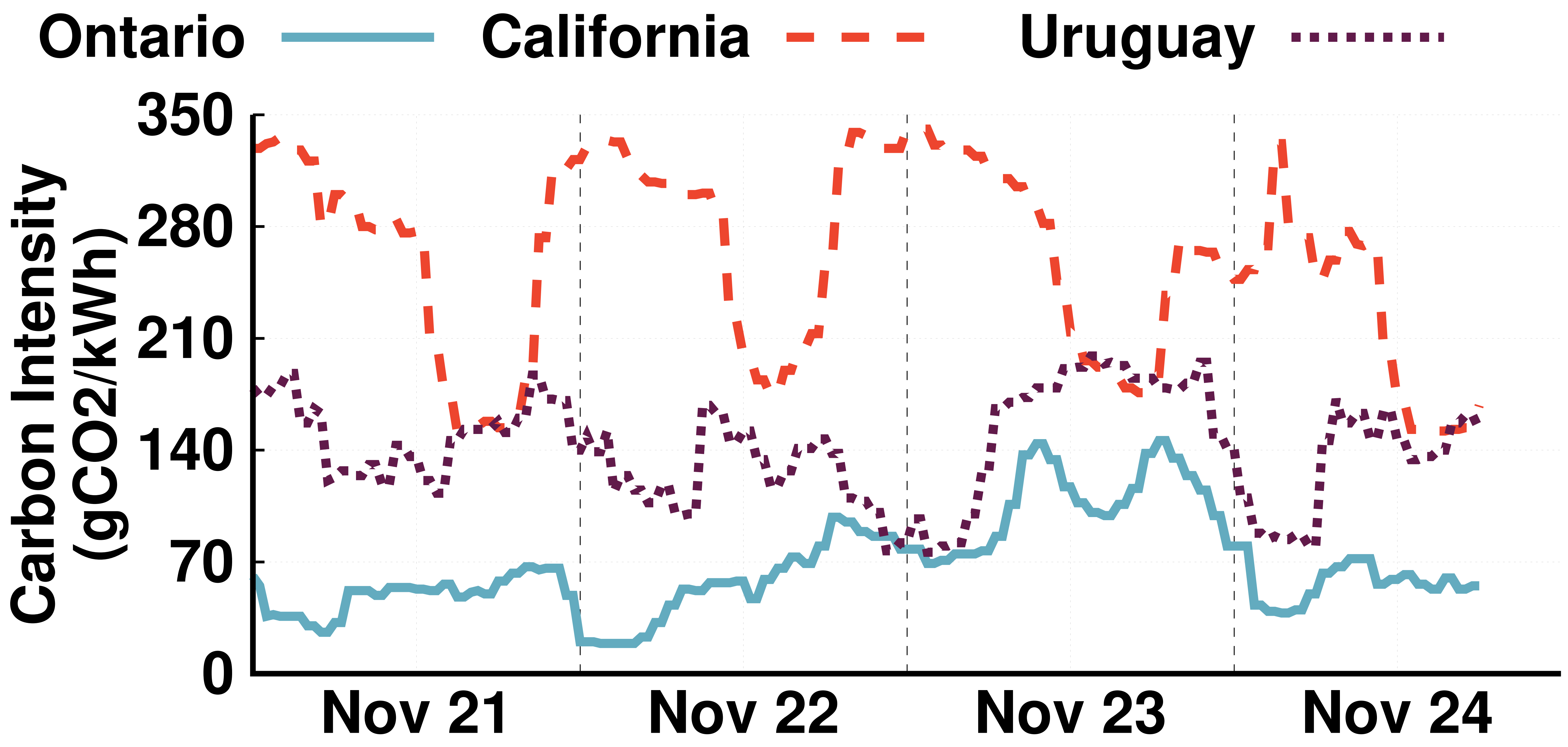
- Operational carbon footprint - **NOT** a solved problems
- Problem is going to **GET WORSE**
- Offsets are good, but **NOT** the panacea
- Embodied and Operational are **NOT** additive
- Operational is under our **DIRECT CONTROL**
- Leverage computing’s flexibility

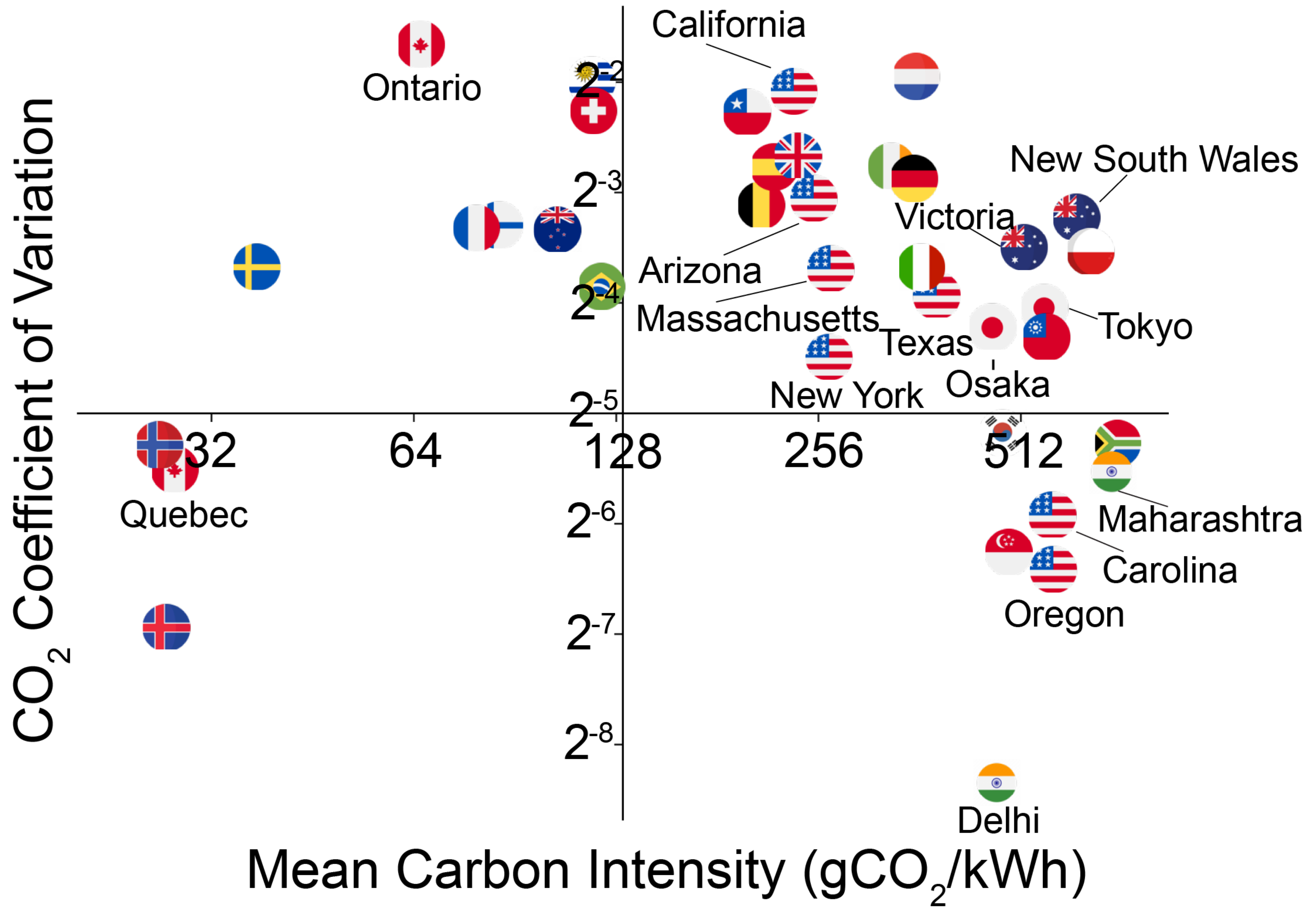
Ecovisor: A Virtual Energy System for Carbon-efficient Applications

Noman Bashir - University of Massachusetts Amherst
To appear at ASPLOS'23

Clean Energy is Variable and Unreliable

- Carbon intensity variation: **less than 50g** to **more than 800g** across time and geographical regions.



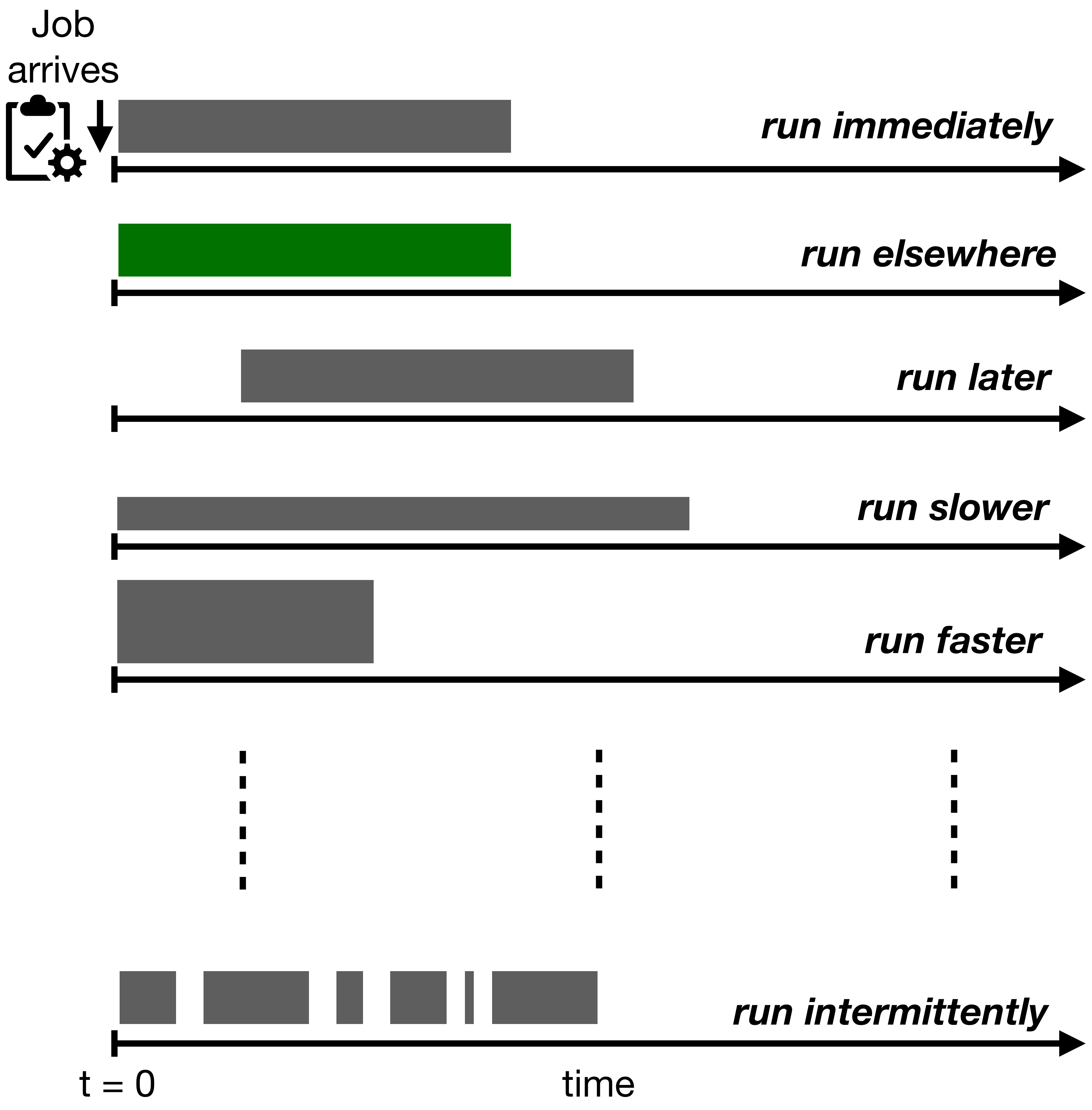


More regions in the world would look like Ontario in near future.

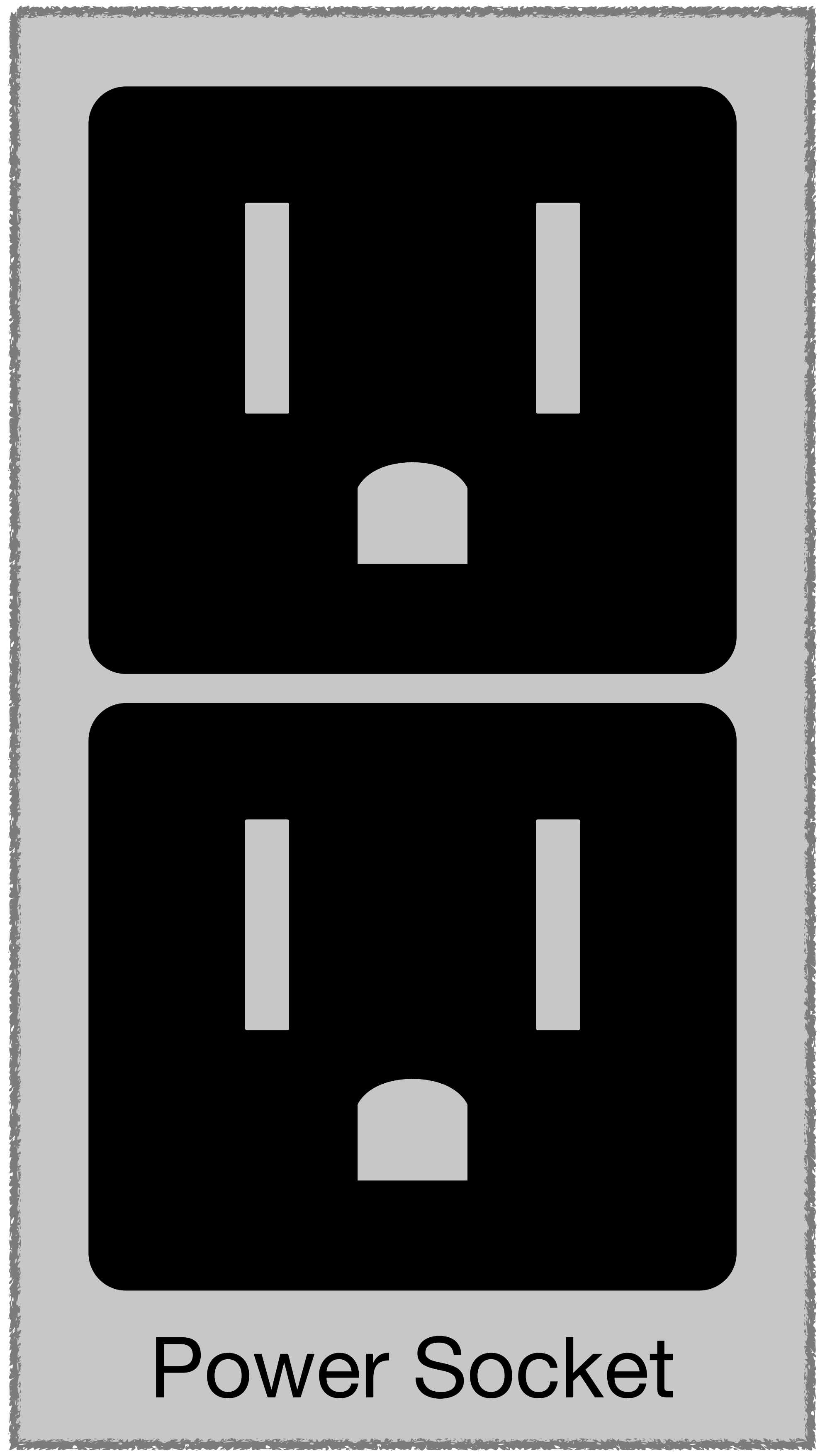
Source: electricityMap

Energy’s Reliability Abstraction Limits Computing’s Potential

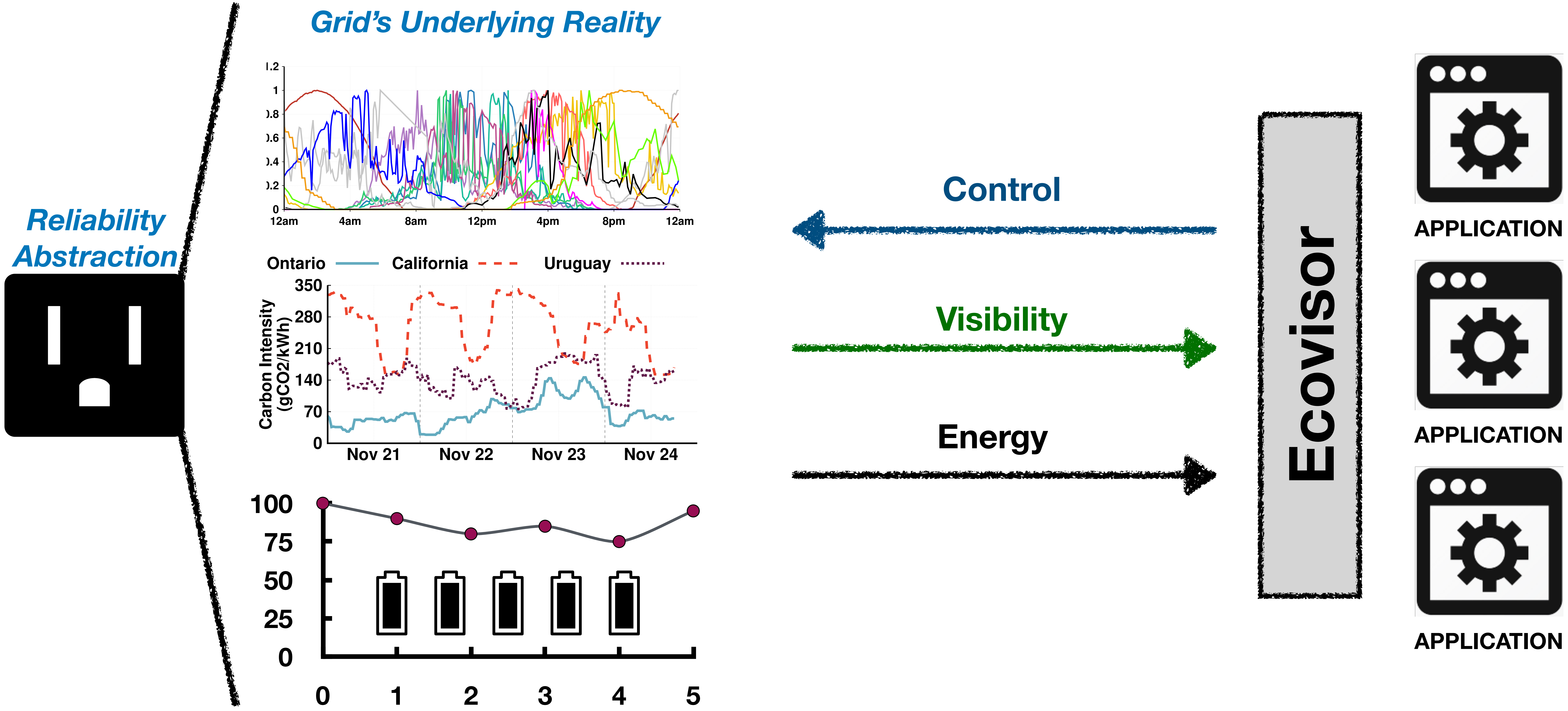
Computing’s Unique Advantages



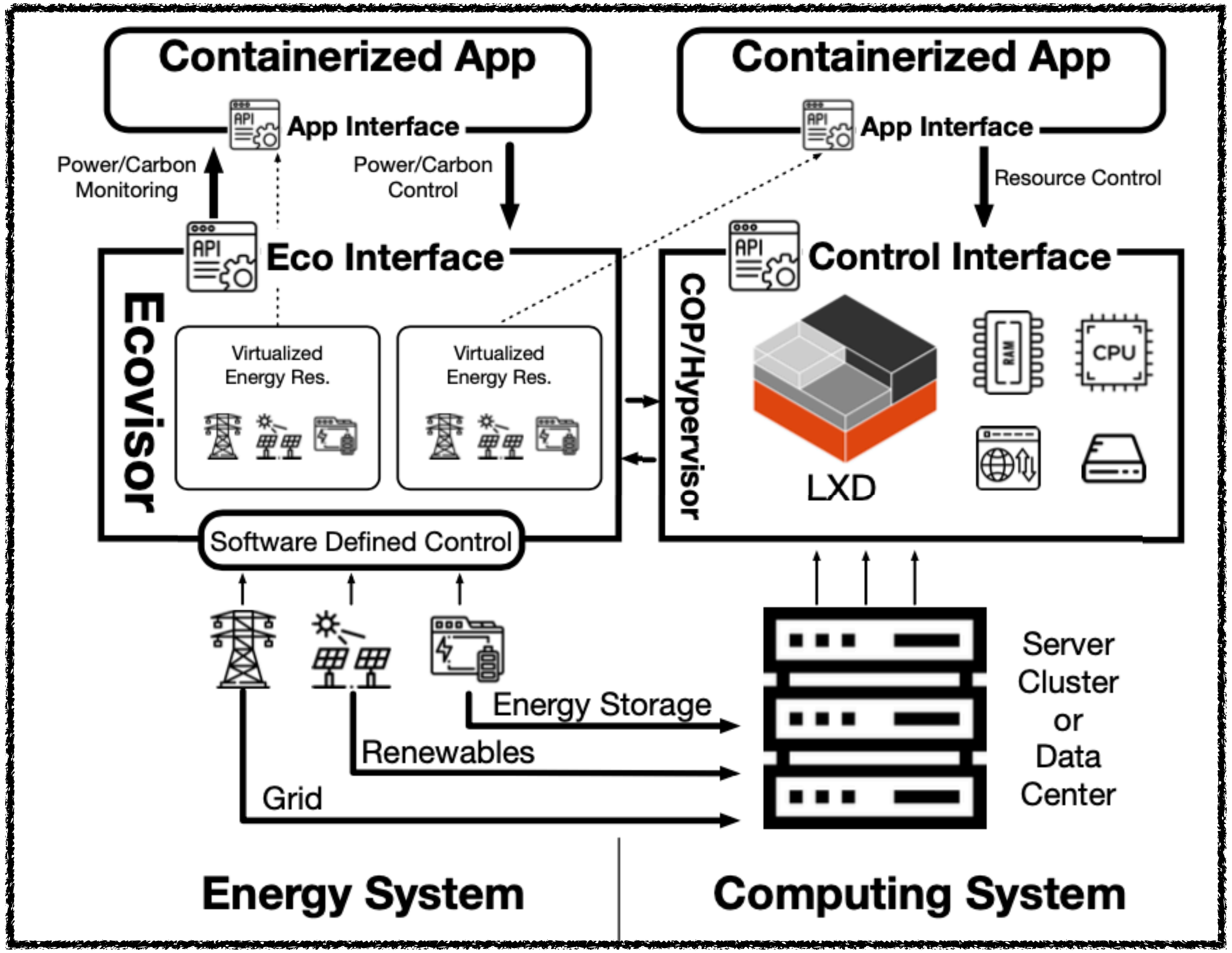
Grid’s Reliability Abstraction



Ecovisor: A Virtual Energy System for Carbon-Efficient Applications



Ecovisor: Design and API



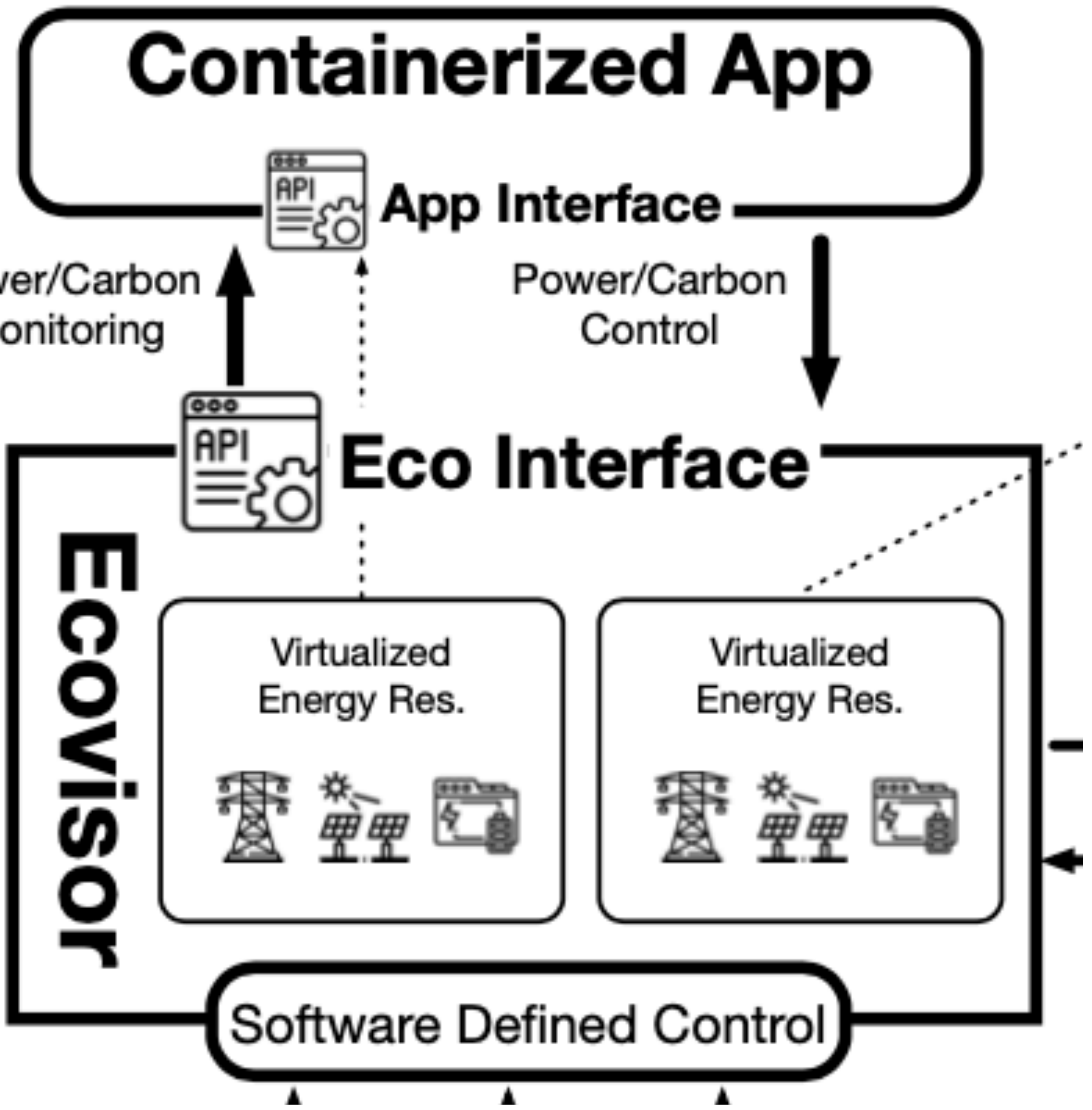
Function Name	Type	Input	Return Value	Description
set_container_powercap()	Setter	ContainerID, kW	N/A	Set a container's power cap
set_battery_charge_rate()	Setter	kW	N/A	Set battery charge rate until full
set_battery_max_discharge()	Setter	kW	N/A	Set max battery discharge rate
get_solar_power()	Getter	N/A	kW	Get virtual solar power output
get_grid_power()	Getter	N/A	kW	Get virtual grid power usage
get_grid_carbon()	Getter	N/A	$g \cdot CO_2/kW$	Get current grid carbon intensity
get_battery_discharge_rate()	Getter	N/A	kW	Get current rate of battery discharge
get_battery_charge_level()	Getter	N/A	kWh	Get energy stored in virtual battery
get_container_powercap()	Getter	ContainerID	kW	Get a container's power cap
get_container_power()	Getter	ContainerID	kW	Get a container's power usage
tick()	Notification	N/A	N/A	Invoked by ecovisor every Δt

Control Power Supply and Demand

Asynchronous Notifications

Get Energy System Information

Ecovisor: Design and API

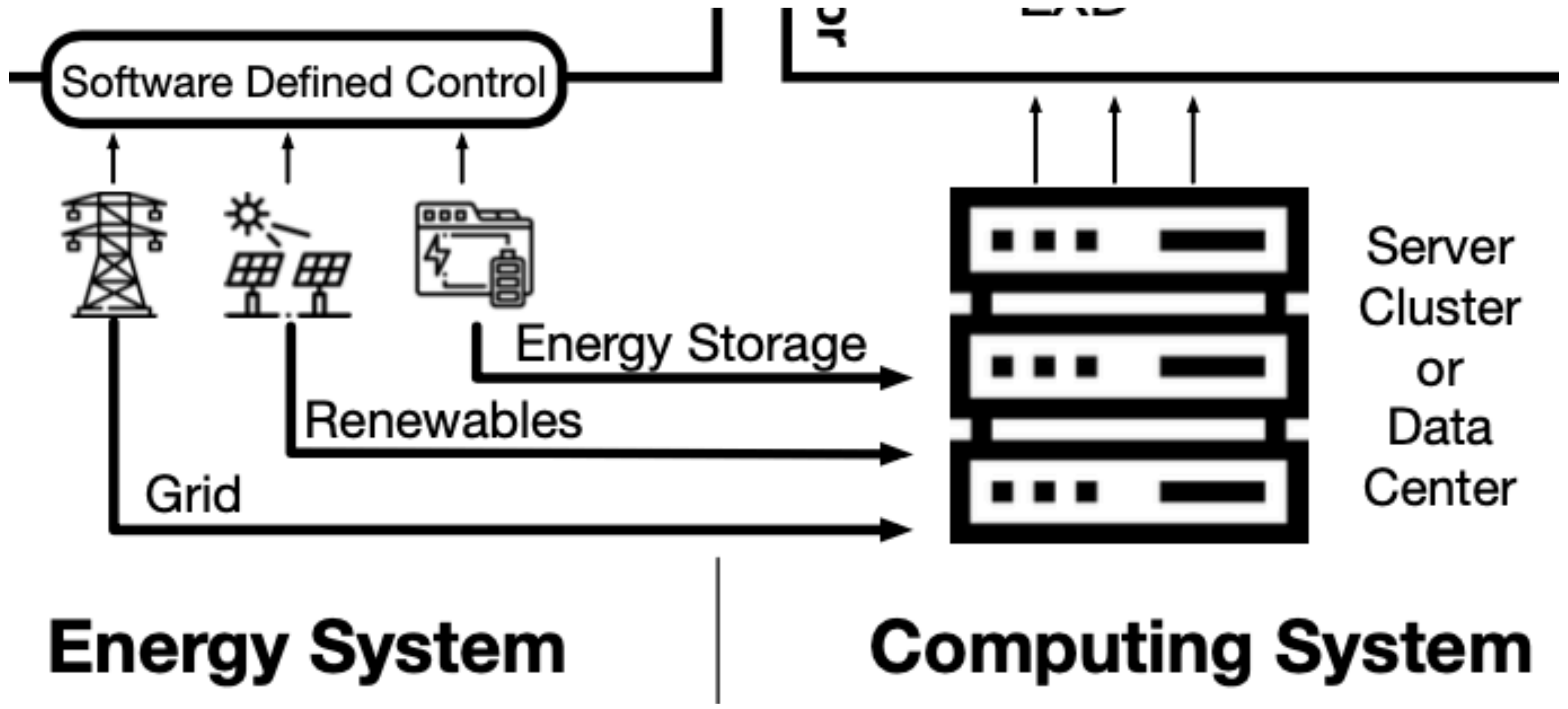


Function Name	Type	Input	Return Value	Description
`set_container_powercap()`	Setter	ContainerID, kW	N/A	Set a container's power cap
`set_battery_charge_rate()`	Setter	kW	N/A	Set battery charge rate until full
`set_battery_max_discharge()`	Setter	kW	N/A	Set max battery discharge rate
`get_solar_power()`	Getter	N/A	kW	Get virtual solar power output
`get_grid_power()`	Getter	N/A	kW	Get virtual grid power usage
`get_grid_carbon()`	Getter	N/A	$g \cdot CO_2/kW$	Get current grid carbon intensity
`get_battery_discharge_rate()`	Getter	N/A	kW	Get current rate of battery discharge
`get_battery_charge_level()`	Getter	N/A	kWh	Get energy stored in virtual battery
`get_container_powercap()`	Getter	ContainerID	kW	Get a container's power cap
`get_container_power()`	Getter	ContainerID	kW	Get a container's power usage
`tick()`	Notification	N/A	N/A	Invoked by ecovisor every Δt

Control Power Supply and Demand

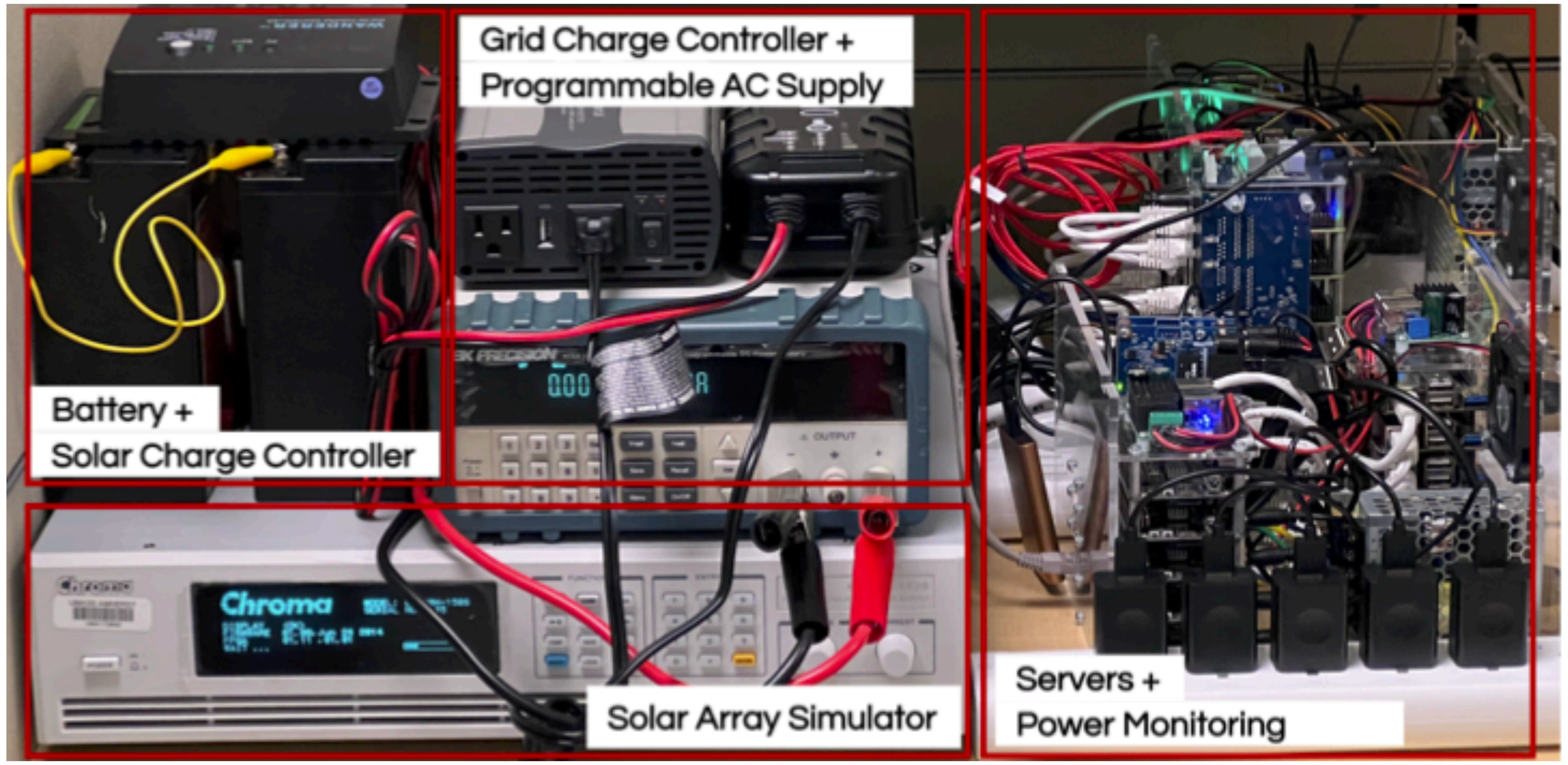
Asynchronous Notifications

Get Energy System Information



Ecovisor: Prototype Implementation

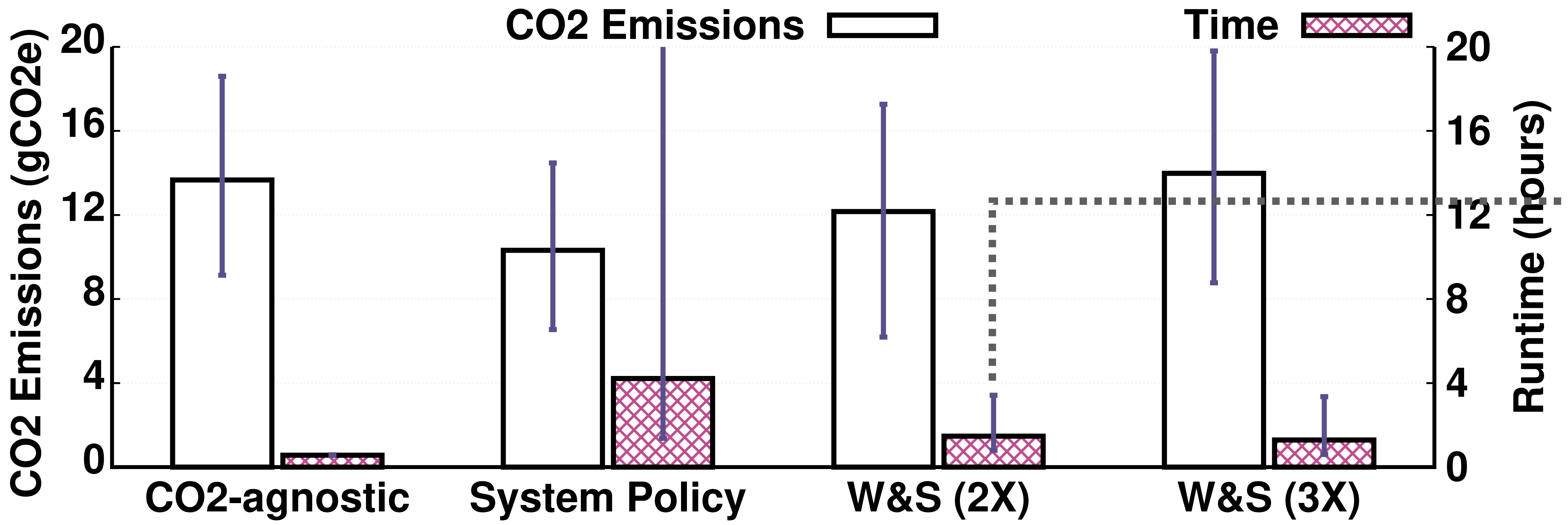
- Software: REST API
  - Access to energy APIs and electricityMap
  - Extends LXD; wraps LXD server
- Hardware: small-scale prototype



- Reducing carbon (ML training, MPI)
  - System (WaitAWhile - Middleware'21)
  - App-specific (Wait&Scale - under review)
- Budgeting carbon (web server)
  - System (rate limiting)
  - App-specific (budgeting)
- Leveraging batteries (web server, Spark)
  - System (static power)
  - App-specific (dynamic power)
- Leveraging solar (MPI, straggler)
  - System (equal)
  - App-specific (progress-based)

Ecovisor: Optimizing Carbon/Performance Trade-off

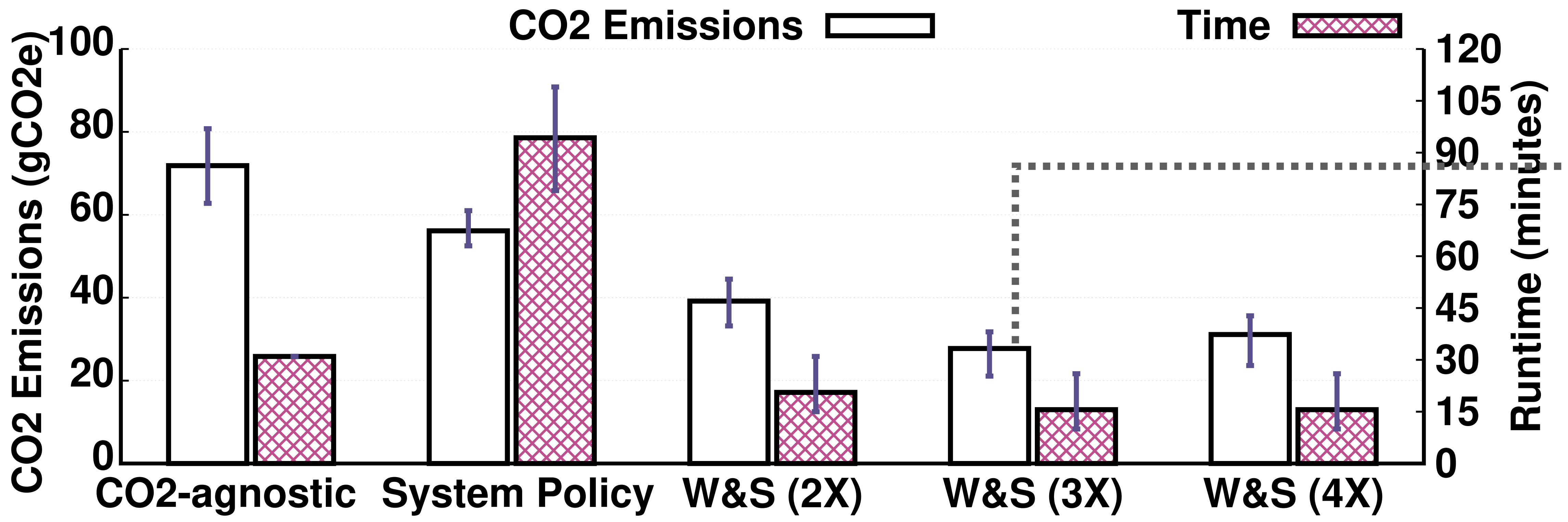
- System (WaitAWhile - Middleware '21) versus Application-specific (Wait&Scale) policy



PyTorch ML Training

Optimal Scale = 2X

Under-review work on leveraging workload elasticity.



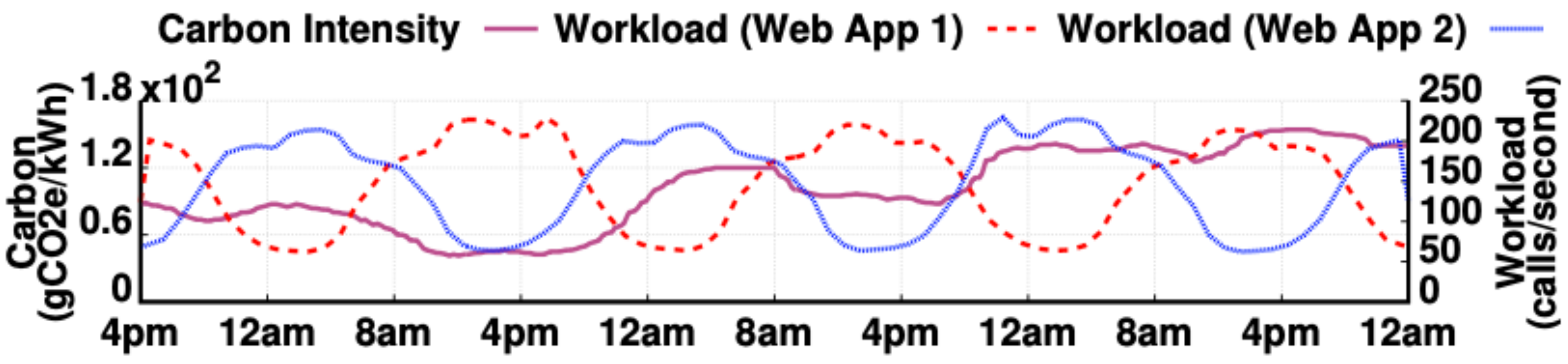
BLAST

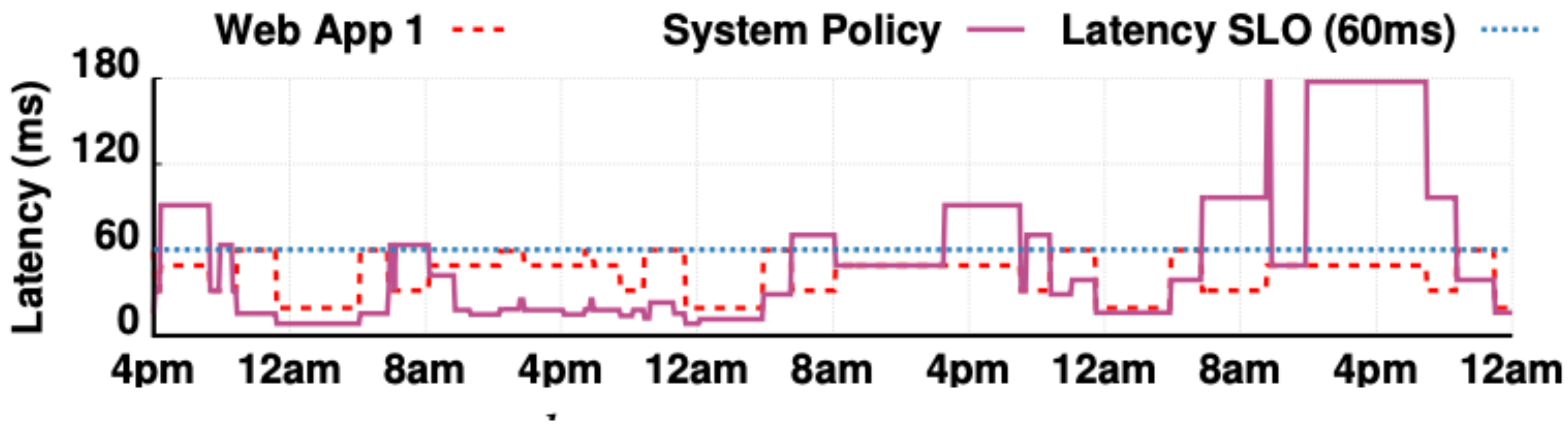
Optimal Scale = 3X

Embarrassingly parallel job.

Ecovisor: Carbon Budgeting

- System (carbon rate-limiting) versus Application-specific (carbon budgeting) policy





Key Point: ***Application-specific carbon budgeting provides useful flexibility***

Conclusion

- **Key Point:** Many carbon-efficiency optimizations possible if applications have visibility/control
- Ecovisor exposes useful functions to enable carbon-efficient applicationsAccess to energy APIs and electricityMap
- A Foundation for developing new abstractions to simplify developing carbon-efficient applications.
- **Ongoing Work:** Exploiting flexibility to reduce carbon; developing new abstractions for ecovisor

Links

- **CarbonFirst:** http://carbonfirst.org/
- **Personal Webpage:** https://noman-bashir.github.io/

Collaborators

- **UMass:** Abel Souza, Walid Hanafy, Qianlin Liang, Jorge Murillo, David Irwin, Prashant Shenoy, Ramesh Sitaraman, Mohammad Hajiesmaili
- **Caltech:** Adam Wierman
- **WPI:** Tian Guo