Let Us Not Put All Our Eggs in One Basket
New Research Directions in Computer Science

Florence Maraninchi
www-verimag.imag.fr/~maraninx

Verimag/Grenoble INP - Ensimag

November 17, 2022

Acknowledgements

CNRS/Ecoinfo, Verimag/ETiCS, CITI/Phenix, Campus d'Après Grenoble, Séminaire transdisciplinaire sur l'anthropocène Grenoble, Inria/STEEP and Inria/Spades, Ensimag MEGA, GDR GPL, ...

General Framework:
Social and Environmental Responsibilities of the Digital World

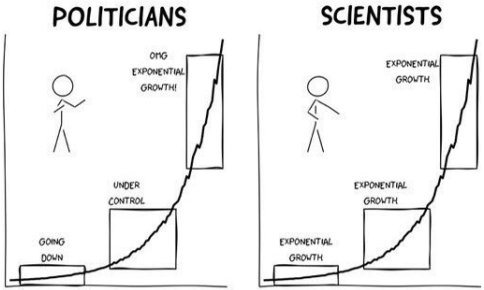
Many topics:

- Generalized surveillance, privacy
- Fragility of the infrastructures, cybersecurity, failures
- Illectronism
- Algorithmic governmentality
- Impacts on the environment

Impacts on the environment

Some figures:

- Between 1.8 and 3.9% of total GHG emissions according to [1]
- Growth rate estimates: 6% per year according to the Shift Project



[1] Charlotte Freitag, Mike Berners-Lee, Kelly Widdicks, Bran Knowles, Gordon S Blair, and Adrian Friday. The real climate and transformative impact of ICT: A critique of estimates, trends, and regulations. Patterns, 2(9):100340, 2021. https://www.sciencedirect.com/science/article/pii/S2666389921001884.

Impacts on the environment and Computer Science

viewpoints

DOI:10.1145/3503916 Bran Knowles, Kelly Widdicks, Gordon Blair, Mike Berners-Lee, and Adrian Friday

Viewpoint

Our House Is On Fire

The climate emergency and computing's responsibility.

WE ARE WRITING this as the world's leaders gather at the UN Climate Change Conference (COP26). In today's news, Boris Johnson is "upbeat," reporting that if this were a football match, the world is down only 5-2 or 5-3, as opposed to 5-1 only a few days earlier. As China's leaders (conspicuously absent) haggle over whether the target should be 2 degrees Celsius warming instead of 1.5 degrees, and nations are more

Impacts on the environment and Computer Science

SIGN IN

COMMUNICATIONS OF THE ACM

Search

HOME | CURRENT ISSUE | NEWS | BLOGS | OPINION | RESEARCH | PRACTICE | CAREERS | ARCHIVE | VI

Home / Magazine Archive / September 2022 (Vol. 65, No. 9) / September 2022 (Vol. 65, No. 9) / Abstract

VIEWPOINT

Let Us Not Put All Our Eggs in One Basket

By Florence Maraninchi
Communications of the ACM, September 2022, Vol. 65 No. 9, Pages 35-37
10.1145/3528088
Comments

VIEW AS: SHARE:

SIGN IN for Full Access

User Name

Password

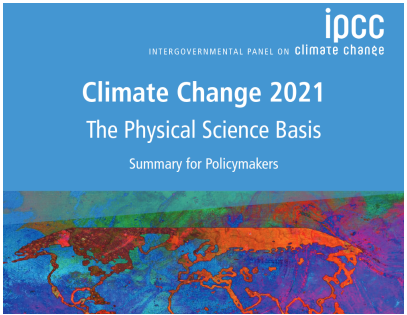
» Forgot Password?

» Create an ACM Web Account

SIGN I

Our colleagues at the Intergovernmental Panel on Climate Change (IPCCa[a]) and the Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem

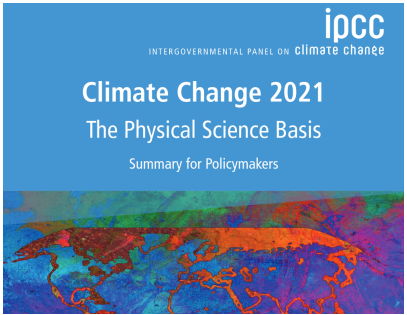
How Should a Researcher in Computer Science React When Reading the IPCC Reports?



Possible answers:

[1] *https://cacm.acm.org/magazines/2020/1/241717-publish-and-perish/fulltext*

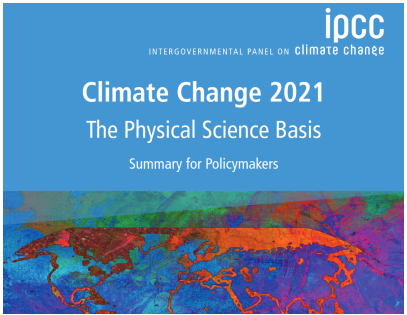
How Should a Researcher in Computer Science React When Reading the IPCC Reports?



Possible answers:
— I don't care

[1] *https://cacm.acm.org/magazines/2020/1/241717-publish-and-perish/fulltext*

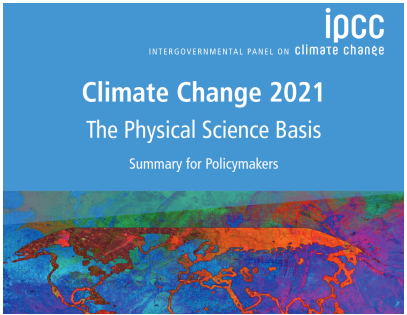
How Should a Researcher in Computer Science React When Reading the IPCC Reports?



Possible answers:
— I don't care
— I do care, but not in my professional life

[1] *https://cacm.acm.org/magazines/2020/1/241717-publish-and-perish/fulltext*

How Should a Researcher in Computer Science React When Reading the IPCC Reports?



Possible answers:
— I don't care
— I do care, but not in my professional life

— No research is neutral, what's my impact? I care also in my professional life: I stopped flying, and I started questioning my research objects.

See also *Publish and Perish - M. Vardi*[1].

[1] *https://cacm.acm.org/magazines/2020/1/241717-publish-and-perish/fulltext*

Working Context vs Research Topics

Working Context :

- GHG estimations for labs
- Personal decisions: stay grounded
- Don't buy a new laptop every 2 years

Research Topics :

- Is the digital world part of the solution?
- How to reduce its impacts?

Working Context vs Research Topics

Working Context :

- GHG estimations for labs
- Personal decisions: stay grounded
- Don't buy a new laptop every 2 years

Research Topics :

- Is the digital world part of the solution?
- How to reduce its impacts?

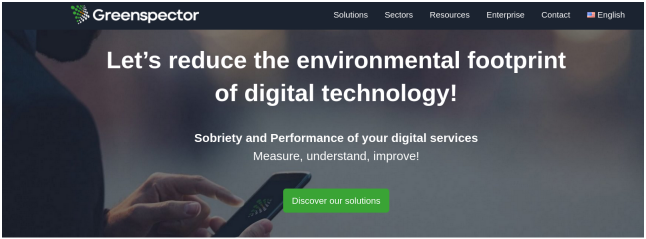
My Background = Critical Embedded Systems : It's better to design simple, non reconfigurable, limited, solutions, but whose correctness is easily understandable and provable. Extensible and sophisticated optimizations that take hours to understand are not a very good idea. I have zero taste for sophistication as an objective.

We're going to talk about research topics.

The Current Green × IT Landscape

- Green IT:
  - Measures/estimations/modeling of (mainly) energy consumption
  - Optimization (SW, HW, communication)
- IT for Green:
  - Optimizations of existing non-IT domains (supply chain, smart-*)
  - New domains (car-sharing platforms)
- Other (e.g., Climate) Sciences and IT:
  - Instrumentation of physical or social phenomena
  - Modeling and simulation

Green IT Example (Greenspector)



A measurement and analysis solution to reduce the environmental impact of mobile and web applications

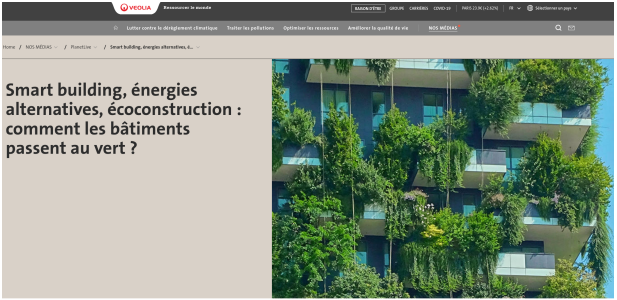
Our expertise:
A unique technological base and robust impact model, the results of +10 years of R&D

The only product on the market based **on real measurements**

2

[2] *https://greenspector.com/en/home/*

Green-by-IT Example

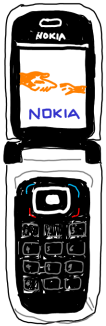


"Pilotage intelligent de la consommation énergétique, valorisation de l'énergie en milieu urbain, végétalisation et biomimétisme. . . Les pistes pour décarboner le secteur du bâtiment font émerger une nouvelle conception de l'énergie, digitalisée et décentralisée."

This Talk

1. Motivations for a New Topic in the Green $\times$ IT Landscape: Limits
2. Rebound Effects, Positive Impacts, Videos of cats, ...
3. Limits as a Computer Science Research Topic
4. This is Not a Conclusion

Typical Situation in 2005



Typical Situation in 2020

Mobile Communications 2005 - 2020

- 2005: Use them to place and receive calls “everywhere”; charge once a week; telephone booths remain;

Mobile Communications 2005 - 2020

- 2005: Use them to place and receive calls "everywhere"; charge once a week; telephone booths remain;
- 2005 ... 2020: Huge improvements of the devices (hardware, software, batteries, screens, casing, ...) + huge improvements of the infrastructure

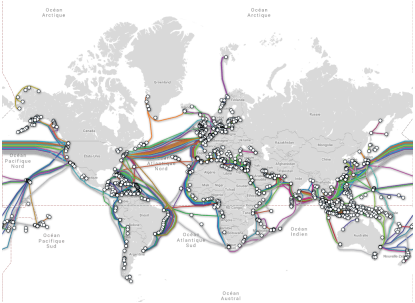
Mobile Communications 2005 - 2020

- 2005: Use them to place and receive calls "everywhere"; charge once a week; telephone booths remain;
- 2005 ... 2020: Huge improvements of the devices (hardware, software, batteries, screens, casing, ...) + huge improvements of the infrastructure
- 2020: Use them mainly as portable always-connected computers; have allowed new services (Uber, maps+GPS, ...); charge twice-a-day or carry an external battery; telephone booths have disappeared; electric charging stations have appeared everywhere (bicycle-powered in railway stations, cafes, ...)

Evolution of the Global Impact of such Mobile Devices and the Underlying Infrastructure?

Both the potential uses and the environmental impacts increased a lot. Is it ok? Do smartphones replace (or rather add up to) something else that also has a very bad impact (laptops, cameras)? How to decide whether optimizations win over rebound effects?

About the Infrastructure



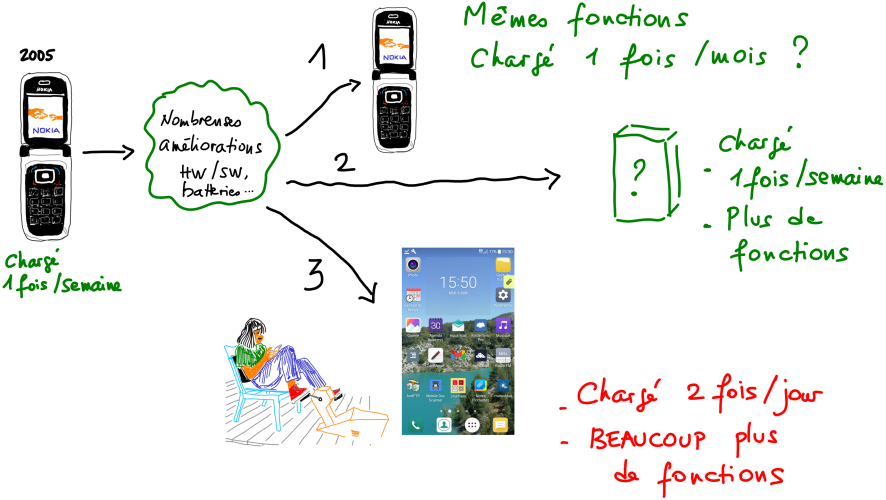
The internet of 2020 is represented by a large block indicating the weight of all the electrons running through it, with a much smaller block showing how it appeared in 2000.

34

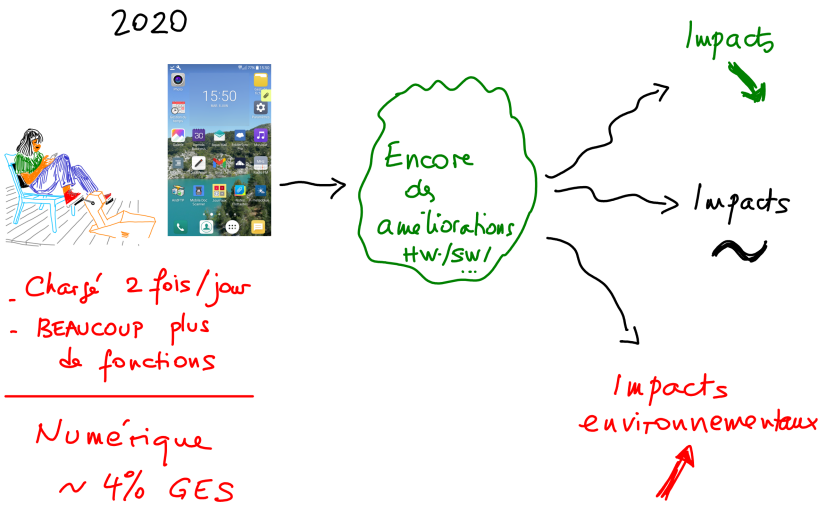
[3] *https://www.theguardian.com/artanddesign/2021/nov/03/art-shed-materialism-fragile-future-technology*

[4] *https://en.wikipedia.org/wiki/Submarine_communications_cable*

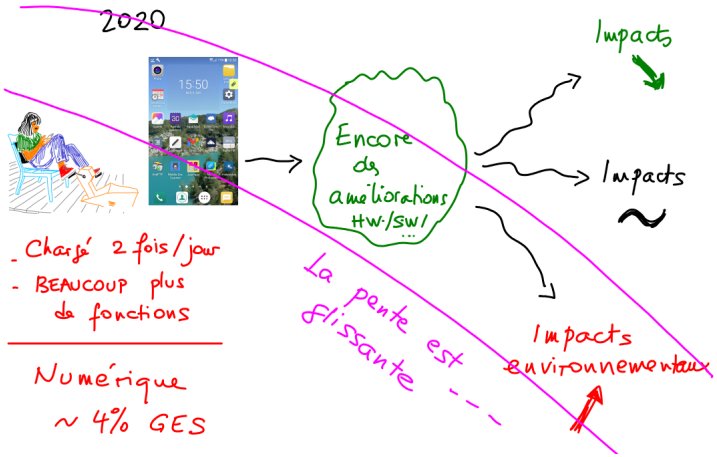
Theoretically Thinkable Paths in 2005



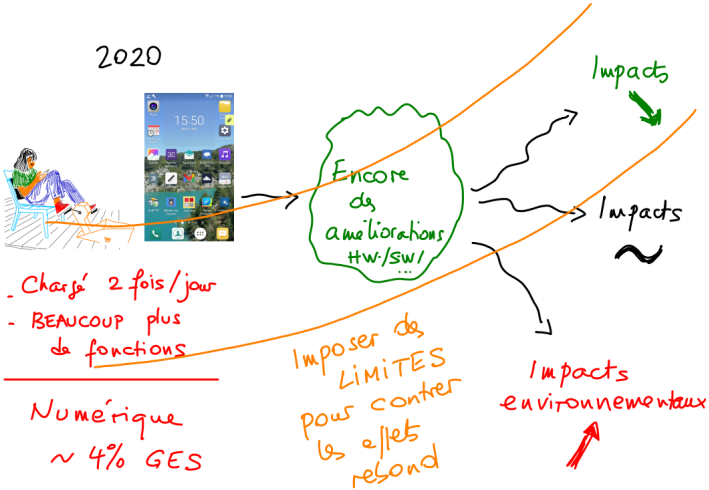
How Do We See The Future, say in 2035?



The Slippery Slope: Towards More Impacts



Imposing Limits to Avoid The Slippery Slope

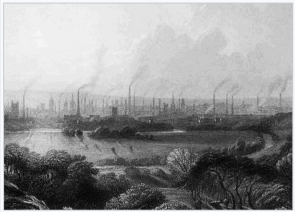


The Jevons Paradox and Rebound Effects

Jevons paradox

From Wikipedia, the free encyclopedia

In economics, the **Jevons paradox** (/ˈdʒɛvənz/; sometimes **Jevons effect**) occurs when technological progress or government policy increases the efficiency with which a resource is used (reducing the amount necessary for any one use), but the falling cost of use increases its demand, negating reductions in resource use.[1] The Jevons' effect is perhaps the most widely known paradox in environmental economics.[2] However, governments and environmentalists[*needs update*] generally assume that efficiency gains will lower resource consumption, ignoring the possibility of the effect arising.[3]



Coal-burning factories in 19th-century Manchester, England. Improved technology allowed coal to fuel the Industrial Revolution, greatly increasing the consumption of coal.

In 1865, the English economist William Stanley Jevons observed that technological improvements that increased the efficiency of coal use led to the increased consumption of coal in a wide range of industries. He argued that, contrary to common intuition, technological progress could not be relied upon to reduce fuel consumption.[4][5]

The issue has been re-examined by modern economists studying consumption rebound effects from improved energy efficiency. In addition to reducing the amount needed for a given use, improved efficiency also lowers the relative cost of using a resource, which increases the quantity demanded. This counteracts (to some extent) the reduction in use from improved efficiency. Additionally, improved efficiency increases real incomes and accelerates economic growth, further increasing the demand for resources. The Jevons' effect occurs when the effect from increased demand predominates, and the improved efficiency results in a faster rate of resource

See Also (in French)

TIC : effets directs, rebond et indirects

Jacques COMBAZ
Jacques.Combaz@univ-grenoble-alpes.fr
ecoinfo.cnrs.fr

Usual Opinions in Computer Science Research

- Let us continue to optimize everything, it can't hurt
- Let us hope that the growth will end
- Ok, the growth won't end, but since the digital world allows to reduce the impact of other sectors in vast proportions, it's worth it.

Usual Opinions in Computer Science Research

- Let us continue to optimize everything, it can't hurt
- Let us hope that the growth will end
- Ok, the growth won't end, but since the digital world allows to reduce the impact of other sectors in vast proportions, it's worth it.

My take: If there is a single example in the history of computing, where a particular optimization has not been accompanied by massive direct and indirect rebound effects, then we should study it extensively in order to try and reproduce it. If there is no such example, then we should stop optimizing.

Social Questions vs Computer Science Research Topics

Individual Behaviors or Regulations? Should we keep the videos of cats?

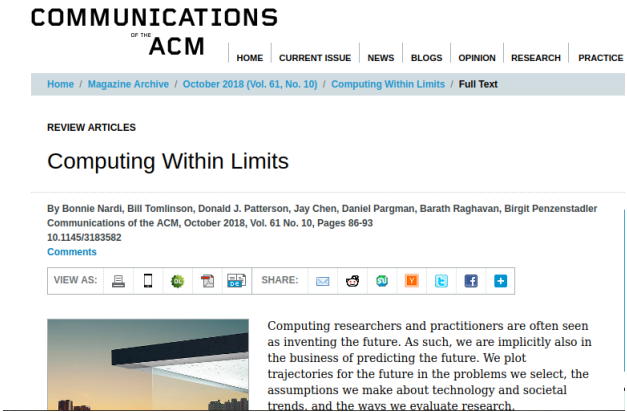
- Even if we don't agree on the need to reduce the uses of digital technologies,
- Even if we don't agree on what to keep/remove,

we can ask intrinsic computer science questions: if we wanted to reduce the impacts of digital technologies (e.g., because of "external" constraints), what could we do, technically?

Some Sources of Inspiration

The enemy of art is the absence of limitations (Orson Welles).
and
The art of computer programming (Donald Knuth).
imply
The enemy of computer programming is the absence of limitations

LIMITS: A Series of Workshops



COMMUNICATIONS OF THE ACM

HOME | CURRENT ISSUE | NEWS | BLOGS | OPINION | RESEARCH | PRACTICE

Home / Magazine Archive / October 2018 (Vol. 61, No. 10) / Computing Within Limits / Full Text

REVIEW ARTICLES

Computing Within Limits

By Bonnie Nardi, Bill Tomlinson, Donald J. Patterson, Jay Chen, Daniel Pargman, Barath Raghavan, Birgit Penzenstadler
Communications of the ACM, October 2018, Vol. 61 No. 10, Pages 86-93
10.1145/3183582
Comments

VIEW AS: SHARE:

Computing researchers and practitioners are often seen as inventing the future. As such, we are implicitly also in the business of predicting the future. We plot trajectories for the future in the problems we select, the assumptions we make about technology and societal trends, and the ways we evaluate research.

6

Motto: *prepare a future of scarcity, in a world of abundant resources*

[6] *https://cacm.acm.org/magazines/2018/10/231374-computing-within-limits/fulltext*

Another Topic: Intrinsically-Limited Digital Systems

Technical approach — for people coming from already-constrained (and somewhat limited) contexts:

- De-Construct to Identify anti-limits (= intrinsically unbounded contexts)
- Rebuild from scratch as a thought-experiment (to escape the tyranny of the state of affairs), self-impose limits on the way

There Are Anti-Limits in a Digital System if it...

- Requires an increasing amount of resources globally (bitcoin alone, or with other crypto-currencies, Chia (proof of space)[7], PKT (proof of bandwidth)[8], NFTs, etc.)

[7] *https://en.wikipedia.org/wiki/Chia_(cryptocurrency)*
[8] *https://pkt.cash/*

There Are Anti-Limits in a Digital System if it...

- Requires an increasing amount of resources globally (bitcoin alone, or with other crypto-currencies, Chia (proof of space)[7], PKT (proof of bandwidth)[8], NFTs, etc.)
- Provisions resources for immediate service delivery, whatever the number of clients

[7] *https://en.wikipedia.org/wiki/Chia_(cryptocurrency)*
[8] *https://pkt.cash/*

There Are Anti-Limits in a Digital System if it...

- Requires an increasing amount of resources globally (bitcoin alone, or with other crypto-currencies, Chia (proof of space)[7], PKT (proof of bandwidth)[8], NFTs, etc.)
- Provisions resources for immediate service delivery, whatever the number of clients
- Assumes unlimited storage — in space and time (initial advertisement of gmail)

[7] *https://en.wikipedia.org/wiki/Chia_(cryptocurrency)*
[8] *https://pkt.cash/*

There Are Anti-Limits in a Digital System if it...

- Requires an increasing amount of resources globally (bitcoin alone, or with other crypto-currencies, Chia (proof of space)[7], PKT (proof of bandwidth)[8], NFTs, etc.)
- Provisions resources for immediate service delivery, whatever the number of clients
- Assumes unlimited storage — in space and time (initial advertisement of gmail)
- Assumes (HW+SW+vendor cloud) availability with no time limit (home automation)

[7] *https://en.wikipedia.org/wiki/Chia_(cryptocurrency)*
[8] *https://pkt.cash/*

There Are Anti-Limits in a Digital System if it...

- Requires an increasing amount of resources globally (bitcoin alone, or with other crypto-currencies, Chia (proof of space)[7], PKT (proof of bandwidth)[8], NFTs, etc.)
- Provisions resources for immediate service delivery, whatever the number of clients
- Assumes unlimited storage — in space and time (initial advertisement of gmail)
- Assumes (HW+SW+vendor cloud) availability with no time limit (home automation)
- (Needs 24/7 connectivity and cloud access) $\times$ (growing number of users)

[7] *https://en.wikipedia.org/wiki/Chia_(cryptocurrency)*
[8] *https://pkt.cash/*

There Are Anti-Limits in a Digital System if it...

- Requires an increasing amount of resources globally (bitcoin alone, or with other crypto-currencies, Chia (proof of space)[7], PKT (proof of bandwidth)[8], NFTs, etc.)
- Provisions resources for immediate service delivery, whatever the number of clients
- Assumes unlimited storage — in space and time (initial advertisement of gmail)
- Assumes (HW+SW+vendor cloud) availability with no time limit (home automation)
- (Needs 24/7 connectivity and cloud access) $\times$ (growing number of users)
- Is built to allow for "unlimited" functional extensions (web)

[7] *https://en.wikipedia.org/wiki/Chia_(cryptocurrency)*
[8] *https://pkt.cash/*

There Are Anti-Limits in a Digital System if it...

- Requires an increasing amount of resources globally (bitcoin alone, or with other crypto-currencies, Chia (proof of space)[7], PKT (proof of bandwidth)[8], NFTs, etc.)
- Provisions resources for immediate service delivery, whatever the number of clients
- Assumes unlimited storage — in space and time (initial advertisement of gmail)
- Assumes (HW+SW+vendor cloud) availability with no time limit (home automation)
- (Needs 24/7 connectivity and cloud access) $\times$ (growing number of users)
- Is built to allow for "unlimited" functional extensions (web)
- Bets on the availability of a better/bigger/more efficient machine, next year (SW obesity)

[7] *https://en.wikipedia.org/wiki/Chia_(cryptocurrency)*
[8] *https://pkt.cash/*

There Are Anti-Limits in a Digital System if it...

- Requires an increasing amount of resources globally (bitcoin alone, or with other crypto-currencies, Chia (proof of space)[7], PKT (proof of bandwidth)[8], NFTs, etc.)
- Provisions resources for immediate service delivery, whatever the number of clients
- Assumes unlimited storage — in space and time (initial advertisement of gmail)
- Assumes (HW+SW+vendor cloud) availability with no time limit (home automation)
- (Needs 24/7 connectivity and cloud access) $\times$ (growing number of users)
- Is built to allow for "unlimited" functional extensions (web)
- Bets on the availability of a better/bigger/more efficient machine, next year (SW obesity)
- Its deployment is profitable only if there are more users and/or increasing usage per user (5G)

[7] *https://en.wikipedia.org/wiki/Chia_(cryptocurrency)*
[8] *https://pkt.cash/*

Candidate Limits (and Induced Constraints)

- Gemini (heavier than gopher, lighter than the web, ...)[9] - no images, no extensions, ...

[9] *https://gemini.circumlunar.space/*
[10] *https://collapseos.org/*

Candidate Limits (and Induced Constraints)

- Gemini (heavier than gopher, lighter than the web, ...)[9] - no images, no extensions, ...
- All configurations in:
  { intermittent, quotas } $\times$ { power, connection, memory, computing power }

[9] *https://gemini.circumlunar.space/*
[10] *https://collapseos.org/*

Candidate Limits (and Induced Constraints)

- Gemini (heavier than gopher, lighter than the web, ...)[9] - no images, no extensions, ...
- All configurations in:
  $\{$ intermittent, quotas $\} \times \{$ power, connection, memory, computing power $\}$
- no centralized architecture, no cloud, no network, no immediate service delivery

[9] *https://gemini.circumlunar.space/*
[10] *https://collapseos.org/*

Candidate Limits (and Induced Constraints)

- Gemini (heavier than gopher, lighter than the web, ...)[9] - no images, no extensions, ...
- All configurations in:
  $\{$ intermittent, quotas $\} \times \{$ power, connection, memory, computing power $\}$
- no centralized architecture, no cloud, no network, no immediate service delivery
- Carefully chosen DSLs (no dynamic allocation, ...) everywhere

[9] *https://gemini.circumlunar.space/*
[10] *https://collapseos.org/*

Candidate Limits (and Induced Constraints)

- Gemini (heavier than gopher, lighter than the web, ...)[9] - no images, no extensions, ...
- All configurations in:
  $\{$ intermittent, quotas $\} \times \{$ power, connection, memory, computing power $\}$
- no centralized architecture, no cloud, no network, no immediate service delivery
- Carefully chosen DSLs (no dynamic allocation, ...) everywhere
- The Ultimate Limit: What if we Stopped Manufacturing New HW Now?

[9] *https://gemini.circumlunar.space/*
[10] *https://collapseos.org/*

Candidate Limits (and Induced Constraints)

- Gemini (heavier than gopher, lighter than the web, ...)[9] - no images, no extensions, ...
- All configurations in:
  { intermittent, quotas } × { power, connection, memory, computing power }
- no centralized architecture, no cloud, no network, no immediate service delivery
- Carefully chosen DSLs (no dynamic allocation, ...) everywhere
- The Ultimate Limit: What if we Stopped Manufacturing New HW Now? *See also "collapse informatics", e.g., CollapseOS*[10]

[9] *https://gemini.circumlunar.space/*
[10] *https://collapseos.org/*

An Example Question (1): is Extensibility a Desirable Property?

> *Extensibility*[11] *is a software engineering and systems design principle that provides for* **future growth.** *Extensibility is a measure of the ability to extend a system and the level of effort required to implement the extension. Extensions can be through the addition of new functionality or through modification of existing functionality. The principle provides for enhancements without impairing existing system functions.*
>
> *An extensible system is one whose internal structure and dataflow are minimally or not affected by new or modified functionality, for example recompiling or changing the original source code might be unnecessary when changing a system's behavior, either by the creator or other programmers. (...)*

Isn't it a slippery slope towards overshoot solutions?

[11] *https://en.wikipedia.org/wiki/Extensibility*

Kubernetes

Kubernetes, also known as K8s, is an open-source system for automating deployment, scaling, and management of containerized applications.

It groups containers that make up an application into logical units for easy management and discovery. Kubernetes builds upon 15 years of experience of running production workloads at Google, combined with best-of-breed ideas and practices from the community.

Planet Scale

Designed on the same principles that allow Google to run billions of containers a week, Kubernetes can scale without increasing your operations team.

Never Outgrow

Whether testing locally or running a global enterprise, Kubernetes flexibility grows with you to deliver your applications consistently and easily no matter how complex your need is.

Extensibility and Lifespan

The longest-lasting computer system I know of is a critical system for nuclear power plants, it's almost 40 years old, and was not extensible at all.

"Pour le blog de mon chat, ai-je vraiment besoin de Kubernetes ? "
For my cat's blog, do I really need Kubernetes?

An Example Question (2): But Where Has All the Memory Gone?

- Think of what it means to manage memory in C, and in Python.
- There is a tendency towards less visibility of the impacts your code can have on the amount of memory.
- It is not necessarily a problem because of the sophisticated garbage collectors coupled with high-level programming languages, but it means the programmer is led to stop worrying about memory use.

Take-Home Messages

- The environmental impacts of digital technologies are already big, and increasing exponentially

Take-Home Messages

- The environmental impacts of digital technologies are already big, and increasing exponentially
- We cannot keep betting on the fact that optimizations will win over rebound effects (it has never been the case)

Take-Home Messages

- The environmental impacts of digital technologies are already big, and increasing exponentially
- We cannot keep betting on the fact that optimizations will win over rebound effects (it has never been the case)
- We cannot keep betting on the fact that digital technologies will reduce the impact of some other sectors in sufficient proportions as to be freed from reduction objectives for themselves

Take-Home Messages

- The environmental impacts of digital technologies are already big, and increasing exponentially
- We cannot keep betting on the fact that optimizations will win over rebound effects (it has never been the case)
- We cannot keep betting on the fact that digital technologies will reduce the impact of some other sectors in sufficient proportions as to be freed from reduction objectives for themselves
- Limiting the digital world requires social changes, computer scientists are not particularly legitimate to drive them, but...

Take-Home Messages

- The environmental impacts of digital technologies are already big, and increasing exponentially
- We cannot keep betting on the fact that optimizations will win over rebound effects (it has never been the case)
- We cannot keep betting on the fact that digital technologies will reduce the impact of some other sectors in sufficient proportions as to be freed from reduction objectives for themselves
- Limiting the digital world requires social changes, computer scientists are not particularly legitimate to drive them, but...
- Even if we don't agree on the need to reduce the uses of digital technologies, even if we don't agree on what to keep/remove, we can ask intrinsic computer science questions: if we wanted to reduce the impacts of digital technologies, **what could we do, technically?**

Let us Not Put All Our Eggs in One Basket

- What if we stopped optimizing (since we are always beated by rebound effects)?
- What if we designed non-extensible systems, on purpose?
- What if we forced the programmers to worry about memory, again?
- Let us start studying limits, just in case...
- Let us prepare to trade "It's convenient" for some hope of sustainability

The End. Thank you.
Questions ?