Ensimag 2A
1. HW Electricity Consumption for SW People
2. Anti-Limits in Digital Systems

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

Verimag/Grenoble INP - Ensimag

2021-2022 – 1st Edition

Résumé

Ce cours est composé de deux parties relativement indépendantes. Des liens apparaîtront pendant l'exposé.

La première partie porte sur la consommation électrique des systèmes informatiques, vue *depuis le logiciel*. A l'Ensimag, grâce aux enseignements d'architecture, systèmes d'exploitation, programmation bas niveau, algorithmique, il est possible de comprendre d'où vient la consommation des systèmes, quels outils sont donnés au programmeur en langage de haut niveau pour la contrôler depuis le logiciel, ou au contraire comment les couches système font en sorte que cela soit transparent pour le programmeur.

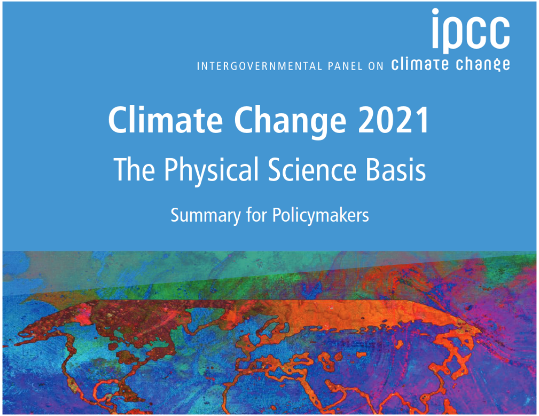
La deuxième partie est basée sur un article "point de vue", à paraître prochainement dans les Communications of the ACM (`https://cacm.acm.org/`). Il s'agit de réfléchir aux principes de conception des systèmes numériques, en se demandant en quoi ils sont limités ou au contraire non-limités et même *anti-limités*.

D'où parlé-je ?

- 30 ans de recherche et enseignement sur les systèmes embarqués temps-réel dur et critiques ; interface HW/SW, propriétés de sûreté, langages de programmation spécialisés et compilation, prototypage virtuel (jumeaux numériques), systèmes contraints (temps, mémoire), développement en temps long, autorités de certification, ...
- Collaborations avec Airbus, STMicroelectronics, OrangeLabs, ...
- Domaines d'application : avionique, ferroviaire, électronique grand public, réseaux de capteurs, villes intelligentes, ...

Ma déformation professionnelle : quand je vois un système informatique je me demande d'abord ce qui pourrait mal se passer.

How Should a Researcher (in CS) React?



ipcc

INTERGOVERNMENTAL PANEL ON climate change

Climate Change 2021

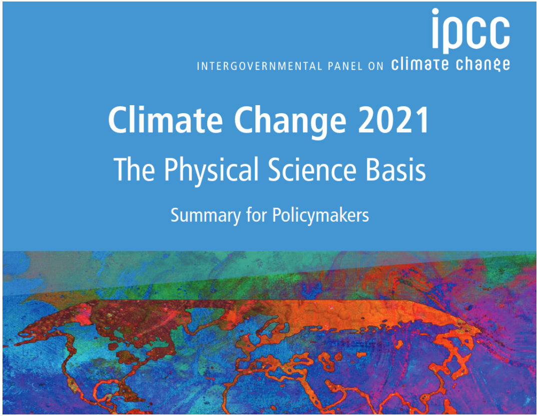
The Physical Science Basis

Summary for Policymakers

Possible answers:

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

How Should a Researcher (in CS) React?



ipcc

INTERGOVERNMENTAL PANEL ON climate change

Climate Change 2021

The Physical Science Basis

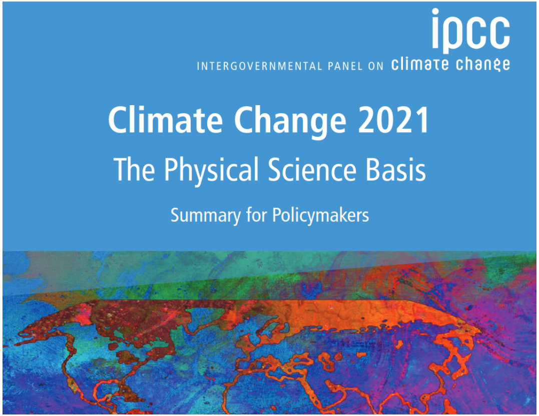
Summary for Policymakers

Possible answers:

— I don't care

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

How Should a Researcher (in CS) React?



ipcc

INTERGOVERNMENTAL PANEL ON climate change

Climate Change 2021

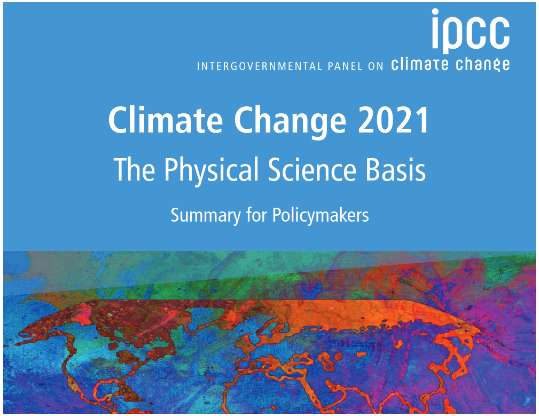
The Physical Science Basis

Summary for Policymakers

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 CS) React?


ipcc
INTERGOVERNMENTAL PANEL ON climate change
Climate Change 2021
The Physical Science Basis
Summary for Policymakers

Possible answers:

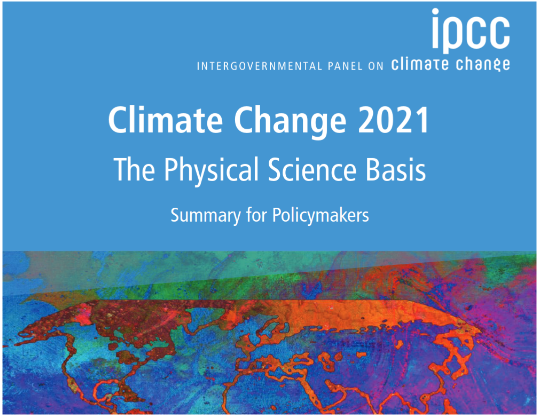
— I don't care

— I do care, but not in my professional life

— I do care, but research is neutral

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

How Should a Researcher (in CS) React?



ipcc

INTERGOVERNMENTAL PANEL ON climate change

Climate Change 2021

The Physical Science Basis

Summary for Policymakers

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

Possible answers:

— I don't care

— I do care, but not in my professional life

— I do care, but research is neutral

— 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.

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

Part I

(HW) Electricity Consumption for SW People

1. The Big Picture: from Physics to Software

2. HW Models for SW People: Timing
   - Time Complexity vs Execution Time
   - “Real-Time” and the 3 Categories of Computer Systems
   - Execution Time: What’s in the HW, What’s in the SW?
     - A Toy Example
     - Real-Life
   - The Notion of Worst-Case Execution Time (WCET)
   - Measuring (Physical) Time as a Security Attack
   - HW Models for WCET Estimation: The Cache Problem
     - Behavior of a Fully-Associative LRU Cache
     - Analyzing the SW

3. HW Models for SW People: Energy Consumption and Temperature
   - Complete Simulation Models
   - Other Topics of Interest
     - RAPL
     - Energy Bugs
     - Simulation Models and the Quest for “Good” Models

Consommation électrique : pourquoi dans ce cours ?

- Parce que dans le thème “informatique et environnement” on parle beaucoup de consommation électrique des équipements, et qu’il est bon de savoir de quoi on parle exactement
- Parce que concevoir des systèmes informatiques en tenant compte de la consommation exige de savoir ce qu’il y a dans une machine (interface HW/SW au moins) et la formation Ensimag le permet
- Parce que souvent les évaluations de consommation reposent sur des modèles, et qu’il faut toujours s’interroger sur les modèles (*tous les modèles sont faux, mais certains sont utiles*) et notre quête des modèles parfaits
- ...

1. The Big Picture: from Physics to Software
2. HW Models for SW People: Timing
3. HW Models for SW People: Energy Consumption and Temperature

From Physics to (Application) Software

Application SW

Decide what to switch on/off

OS

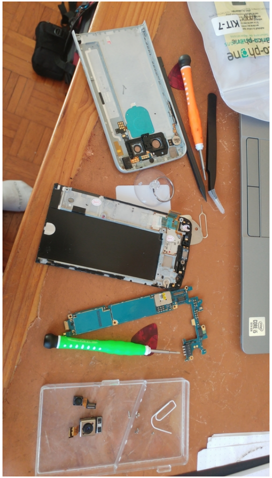
control sleep modes
real–time scheduling and adjusting V, F

Components' operational modes
Power Domains and DVFS

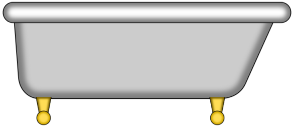
Temperature Sensors

Static+Dynamic Energy Consumption

Battery behaviour and Discharge time



Discharge time is not a Simple Function of Power Consumption (or, *a battery is not a bathtub*)



see "rate-dependency effect" in David Linden et Thomas B. Reddy — Handbook of batteries. McGraw-Hill 2002

Ravishankar Rao, Sarma Vrudhula et Naehyuck Chang — Battery optimization vs energy optimization: which to choose and when?. ICCAD'05

Sources of Power Consumption

$$P = \begin{array}{l} P_{\text{static}} \; \textit{due to leakage currents} \quad + \\ P_{\text{dynamic}} \; \textit{due to the switching of transistors} \end{array}$$

$$P_{\text{static}} = V \times K_1 \times g(T) \quad \nearrow \; \text{when transistor size} \; \searrow$$
$$P_{\text{dynamic}} = F \times V^2 \times \alpha \times K_2$$

V: Voltage, F: Frequency, T: Temperature

g: increasing function

α: activity ratio, or amount of computation performed

K_is: various "constants" depending on the module area and on the synthesis technology

Power Control in Modern Circuits

- Clock Gating (turn off the clock):
  $P_{\text{dynamic}} = 0$, but P_{static} unchanged
- Dynamic Voltage and Frequency Scaling (DVFS) reduces V and F. A circuit can have a (small) number of operating points (V, F). Switching between them has a cost.
- Power Gating (switch a component on/off); Switching is very costly (save/restore state); application-level information is needed (e.g., GPS is not longer used, switch the sub-circuit off).

Complex Feedback Interactions

Consumption ↑

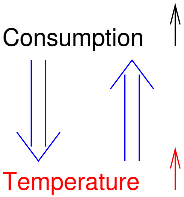
Complex Feedback Interactions

Consumption ↑

⇓

Temperature ↑

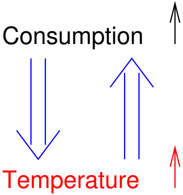
Complex Feedback Interactions



Consumption

Temperature

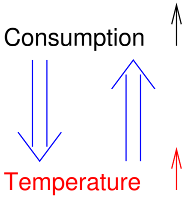
Complex Feedback Interactions



Consumption ↑

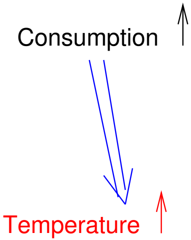
Temperature ↑

Complex Feedback Interactions



Consumption ↑

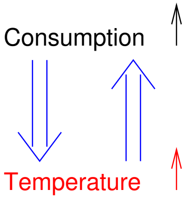
Temperature ↑



Consumption ↑

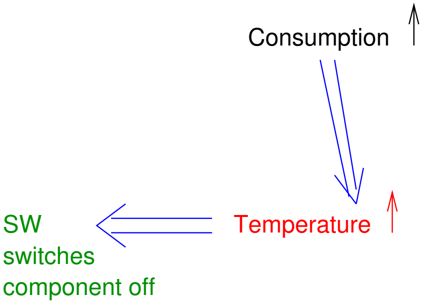
Temperature ↑

Complex Feedback Interactions



Consumption ↑

Temperature ↑

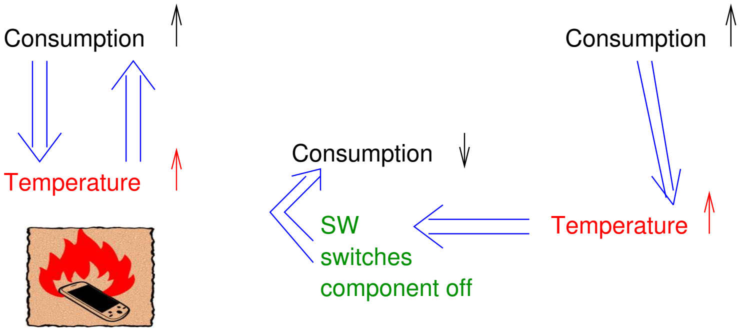


Consumption ↑

Temperature ↑

SW switches component off

Complex Feedback Interactions



Consumption ↑

Temperature ↑

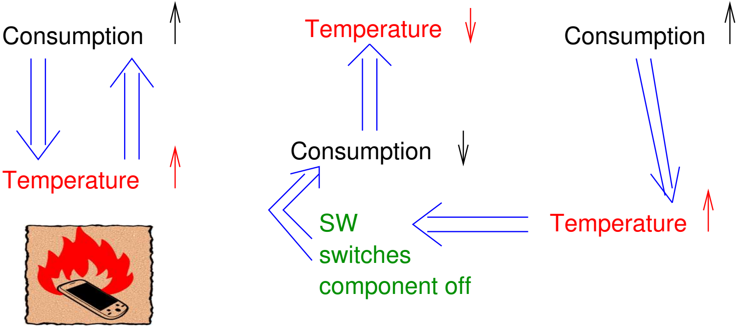
Consumption ↓

SW switches component off

Temperature ↑

Consumption ↑

Complex Feedback Interactions



Consumption ↑

Temperature ↑

Temperature ↓

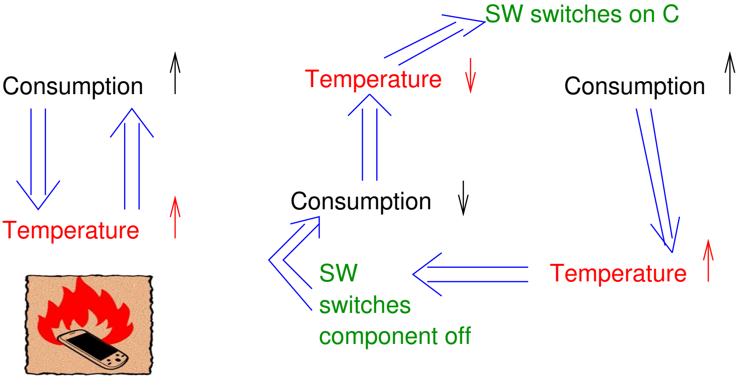
Consumption ↓

SW switches component off

Consumption ↑

Temperature ↑

Complex Feedback Interactions



Consumption ↑

Temperature ↑

Temperature ↓

SW switches on C

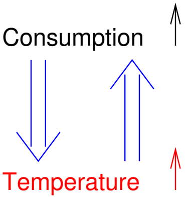
Consumption ↑

Temperature ↑

SW switches component off

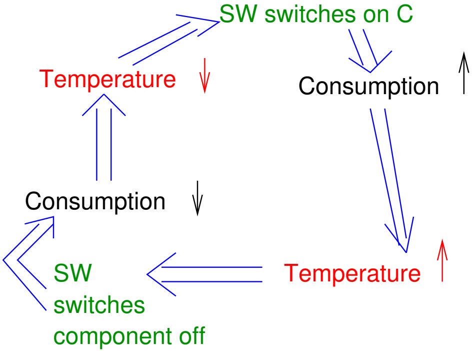
Consumption ↓

Complex Feedback Interactions



Consumption ↑

Temperature ↑



SW switches on C

Consumption ↑

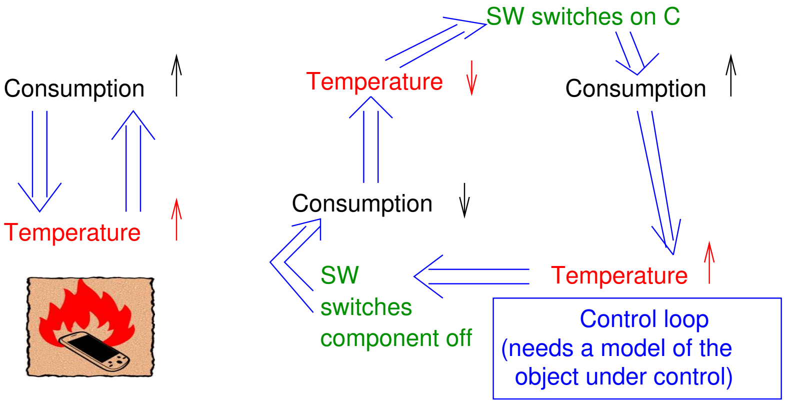
Temperature ↑

SW switches component off

Consumption ↓

Temperature ↓

Complex Feedback Interactions



Consumption ↑

Temperature ↑

SW switches on C

Temperature ↓

Consumption ↑

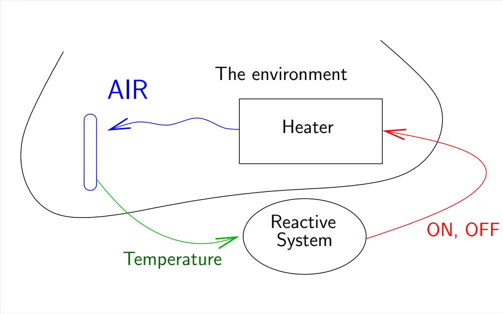
Consumption ↓

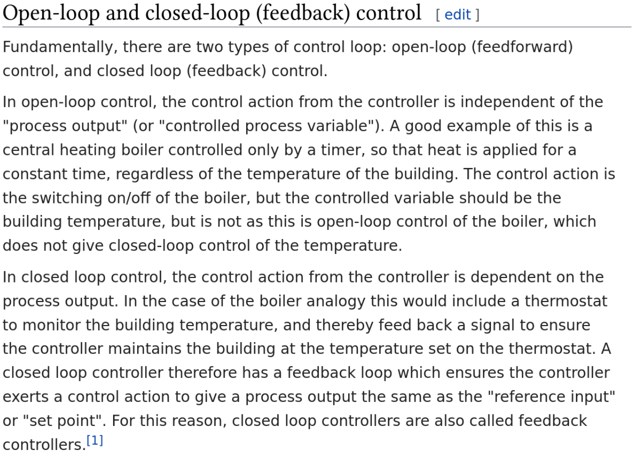
SW switches component off

Temperature ↑

Control loop
(needs a model of the object under control)

Control Loops[2]





Open-loop and closed-loop (feedback) control [edit]

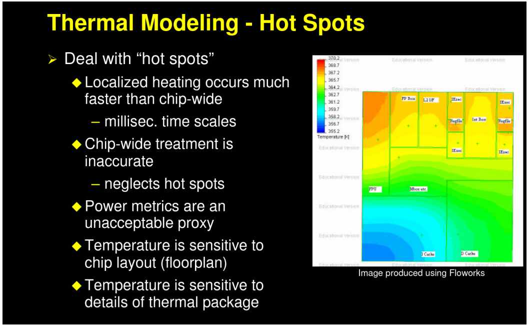
Fundamentally, there are two types of control loop: open-loop (feedforward) control, and closed loop (feedback) control.

In open-loop control, the control action from the controller is independent of the "process output" (or "controlled process variable"). A good example of this is a central heating boiler controlled only by a timer, so that heat is applied for a constant time, regardless of the temperature of the building. The control action is the switching on/off of the boiler, but the controlled variable should be the building temperature, but is not as this is open-loop control of the boiler, which does not give closed-loop control of the temperature.

In closed loop control, the control action from the controller is dependent on the process output. In the case of the boiler analogy this would include a thermostat to monitor the building temperature, and thereby feed back a signal to ensure the controller maintains the building at the temperature set on the thermostat. A closed loop controller therefore has a feedback loop which ensures the controller exerts a control action to give a process output the same as the "reference input" or "set point". For this reason, closed loop controllers are also called feedback controllers.[1]

[2] https://en.wikipedia.org/wiki/Open-loop_controller

Temperature Models



Thermal Modeling - Hot Spots

- Deal with "hot spots"
  - Localized heating occurs much faster than chip-wide
    - millisec. time scales
  - Chip-wide treatment is inaccurate
    - neglects hot spots
  - Power metrics are an unacceptable proxy
  - Temperature is sensitive to chip layout (floorplan)
  - Temperature is sensitive to details of thermal package

370.2
368.7
367.2
365.7
364.2
362.7
361.2
359.7
358.2
356.7
355.2
Temperature [K]
FP Box
L2 I/F
IExec
Regfile
Int Box
IExec
Regfile
IExec
IExec
FPU
Mbox etc.
I Cache
D Cache

Image produced using Floworks

`http://lava.cs.virginia.edu/HotSpot/`

K. Skadron, M. R. Stan, W. Huang, S. Velusamy, K. Sankaranarayanan, and D. Tarjan. Temperature-Aware Microarchitecture. In Proceedings of the 30th International Symposium on Computer Architecture, June 2003.

1. The Big Picture: from Physics to Software

2. HW Models for SW People: Timing
   - Time Complexity vs Execution Time
   - “Real-Time” and the 3 Categories of Computer Systems
   - Execution Time: What’s in the HW, What’s in the SW?
   - The Notion of Worst-Case Execution Time (WCET)
   - Measuring (Physical) Time as a Security Attack
   - HW Models for WCET Estimation: The Cache Problem

3. HW Models for SW People: Energy Consumption and Temperature

2 HW Models for SW People: Timing

- Time Complexity vs Execution Time
- “Real-Time” and the 3 Categories of Computer Systems
- Execution Time: What’s in the HW, What’s in the SW?
- The Notion of Worst-Case Execution Time (WCET)
- Measuring (Physical) Time as a Security Attack
- HW Models for WCET Estimation: The Cache Problem

Time Complexity of Algorithms

Example of sorting algorithms (for an array of n elements):
— bubble sort has time complexity n^2
— heap sort has time complexity $n \log n$.

See also *space complexity*.

Just counting elementary operations.
Relation with time as perceived by human beings is implicit (and sequential here, in the RAM Model; other models exist, e.g., the PRAM[3] Model, for parallel architectures).

[3] https://fr.wikipedia.org/wiki/Parallel_random_access_machine

PRAM

Cf. `https://fr.wikipedia.org/wiki/Parallel_random_access_machine`

> *PRAM modélise une machine parallèle à une mémoire RAM partagée par un ensemble de processeurs. Ces processeurs sont synchronisés par chaque instruction.*
>
> *(...)*
>
> *PRAM ne tient toutefois aucun compte des coûts d'échanges de données entre différentes machines. Notamment, la représentation par PRAM d'une grappe d'ordinateurs, où la mémoire disponible est en réalité partagée entre chaque ordinateur (sic!), négligera le temps d'accès d'un processeur à une partie de la mémoire qui ne lui est pas physiquement locale.*

Execution Time of Programs (1)

```
jorasses(13) /usr/bin/time -p  du -sh ~
21G /home/maraninx
real 92.73
user 1.33
sys 9.59
jorasses(14) /usr/bin/time -p sleep 3
real 3.00
user 0.00
sys 0.00
```

What does “real time” (resp. “user”, “sys”) mean here?

Execution Time of Programs (2)

```
#include <stdio.h>
#include <time.h>

int main () {
  int i;
  clock_t start = clock();

  for (i=0;i<10000000; i++) putchar(’.’);

  clock_t stop = clock();
  double elapsed =
      ((double)(stop - start) * 1000.0 )/
      CLOCKS_PER_SEC;
  printf("\n Time elapsed in ms: %f\n", elapsed);
}
```

Execution Time of Programs (2')

Several executions show significant variations (not a way to impose precise timing!):

```
jorasses(59) /usr/bin/time -p ./spend_time
....................................................................................
 Time elapsed in ms: 147.620000
real 0.65
user 0.11
sys 0.03

jorasses(59) /usr/bin/time -p ./spend_time
....................................................................................
 Time elapsed in ms: 152.287000
real 0.60
user 0.12
sys 0.02
```

Execution Time of Programs (2”)

```
jorasses(62) ./spend_time | grep Time
 Time elapsed in ms: 83.799000
jorasses(63) ./spend_time | grep Time
 Time elapsed in ms: 79.817000
```

2 HW Models for SW People: Timing

- Time Complexity vs Execution Time
- “Real-Time” and the 3 Categories of Computer Systems
- Execution Time: What’s in the HW, What’s in the SW?
- The Notion of Worst-Case Execution Time (WCET)
- Measuring (Physical) Time as a Security Attack
- HW Models for WCET Estimation: The Cache Problem

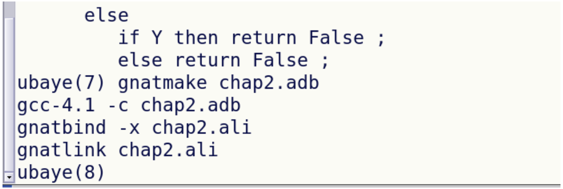
3 Categories of Computer Systems
Towards a definition of real-time systems

- Transformational Systems (Typical example : a compiler)
- Interactive Systems (Typical example: a man-machine interface)
- Reactive Systems (Typical example: a heater controller)

Of course, no complete system is purely of one kind only.

Transformational Systems

Typical example : a compiler



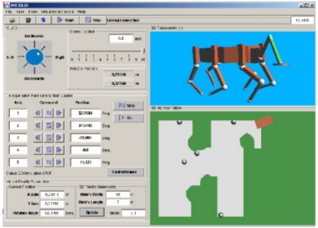
```
      else
         if Y then return False ;
         else return False ;
ubaye(7) gnatmake chap2.adb
gcc-4.1 -c chap2.adb
gnatbind -x chap2.ali
gnatlink chap2.ali
ubaye(8)
```

Inputs at the beginning, then some finite-time computation, outputs at the end.

A transformational system has to terminate.
If it does not, it’s a bug.
The time it takes to complete is a matter of performance (only).

Interactive Systems

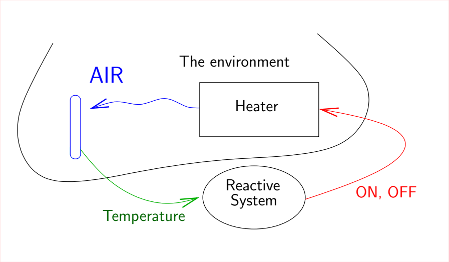
Typical example: a man-machine interface



loop-based behavior (does not necessarily terminate), where inputs come all the time (human actions on buttons, mouse, keyboard) and outputs are produced all the time also (changes of the interface, effects on the underlying computer system).

Reactive Systems

Typical example: a heater controller.



The same as interactive systems, but the speed of the interaction is driven by the (physical) environment. The computer system should be sufficiently fast in order not to miss relevant evolutions of the environment.

Reactivity to an Environment, Real-Time

A typical real-time program:

```
initializations
while (true) {
      --- point (1)
      get inputs
           (read sensors)
      compute outputs
           and update memory
      write outputs
           on the actuators
      --- point (2)
}
```

The time it takes to execute the code between points (1) and (2) is the time between two samples of the inputs. This is real time.

The outputs to the environment may have some influence on future inputs. This is reactivity.

Implementation of Real-Time Systems: Problems

- Write code that is sufficiently fast (you're not always allowed to answer: "*try a bigger machine*" because the HW may be imposed by other concerns)
- Be able to tell how fast your program is, *in advance* (Worst-Case-Execution-Time static evaluation)

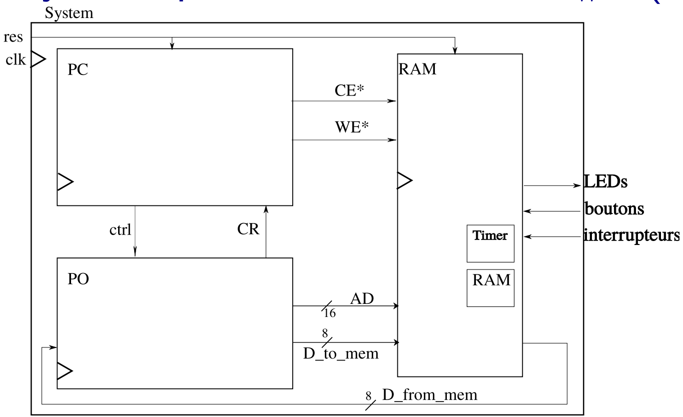
2 HW Models for SW People: Timing

- Time Complexity vs Execution Time
- "Real-Time" and the 3 Categories of Computer Systems
- Execution Time: What's in the HW, What's in the SW?
- The Notion of Worst-Case Execution Time (WCET)
- Measuring (Physical) Time as a Security Attack
- HW Models for WCET Estimation: The Cache Problem

2 HW Models for SW People: Timing

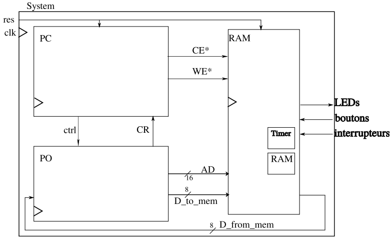
- Time Complexity vs Execution Time
- "Real-Time" and the 3 Categories of Computer Systems
- Execution Time: What's in the HW, What's in the SW?
  - A Toy Example
  - Real-Life
- The Notion of Worst-Case Execution Time (WCET)
- Measuring (Physical) Time as a Security Attack
- HW Models for WCET Estimation: The Cache Problem

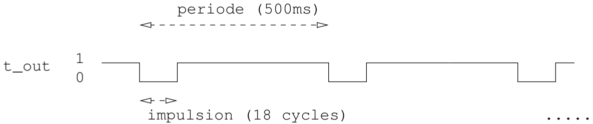
Toy Example - Archi 1A — TP #5 (1)



Simple processor, control and operative parts (PC/PO), plus memory.
Instructions are 3- or 5-cycle long.

Toy Example - Archi 1A — TP #5 (2)



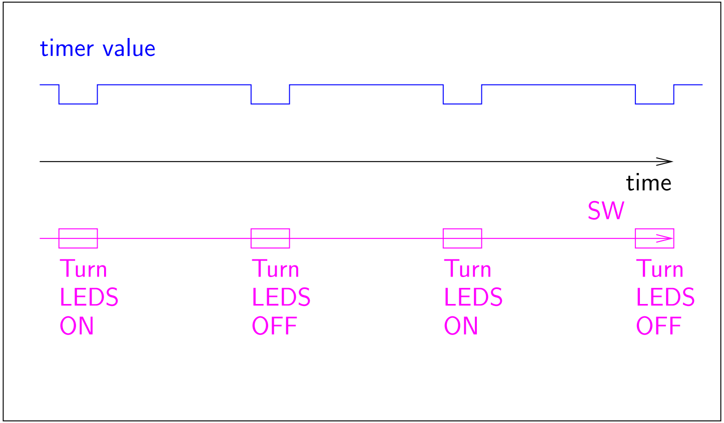


The processor

An external timer

Exercice: write a program that reads the timer (using a read-from-memory instruction, which gives 0 or 1) at appropriate instants so as to be able to switch LEDS on/off at frequency 1 Hz.

Toy Example - Archi 1A — TP #5 (3)
What the SW should do

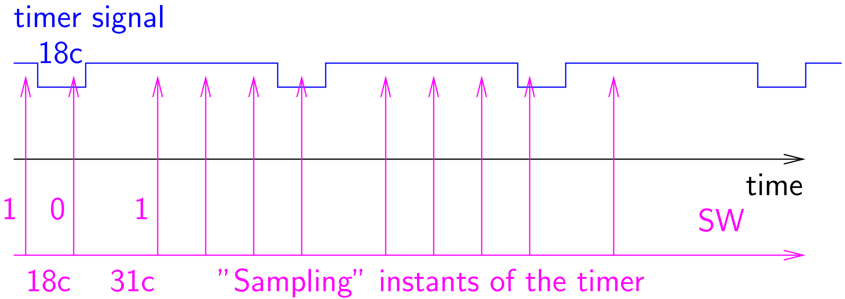


Toy Example - Archi 1A — TP #5 (4)
The SW

```
li 255,r3               // to be displayed by the LEDS
waiting:
 ld 0x0010, r0          // read the timer current value
 and r0,r0              // update bit Z accordingly
 jz switch              // if active, play with the LEDS
 jmp waiting            // otherwise loop to re-check timer
switch:
 not r3                 // invert value to be displayed
 st r3, 0x0001          // display
 jmp waiting            // wait again
```

Questions: determine the delay between two successive reads of the timer. Is it constant? Does this solution work?

Toy Example - Archi 1A — TP #5 (5) What Happens



— `ld 0x0010,r0; and r0,r0; jz switch; jmp waiting` = 18 cycles

— `same; not r3; st r3,0x0001; jmp waiting` = 31 cycles

Toy Example - Archi 1A — TP #5 (6)

Conclusion: this particular program works as intended, thanks to a quite tricky cooperation of the HW and this particular SW program.

Not a very good idea.
Cannot be made sufficiently general.

And...

2 HW Models for SW People: Timing

- Time Complexity vs Execution Time
- “Real-Time” and the 3 Categories of Computer Systems
- Execution Time: What's in the HW, What's in the SW?
  - A Toy Example
  - Real-Life
- The Notion of Worst-Case Execution Time (WCET)
- Measuring (Physical) Time as a Security Attack
- HW Models for WCET Estimation: The Cache Problem

Real-Life: HW is Muuuuuuuuuuuuuch more complicated than this!

All modern processors have features that improve average-case execution time. It's far from predictable. Writing the appropriate sequence of instructions in order to produce a precise timing is hopeless.

Some optimizations:

- Virtual Memory
- Speculative Execution
- Pipelining
- Hyperthreading
- ...

Good old Motorola 6502: one instruction $=x$ cycles; 1 cycle $=n$ seconds.

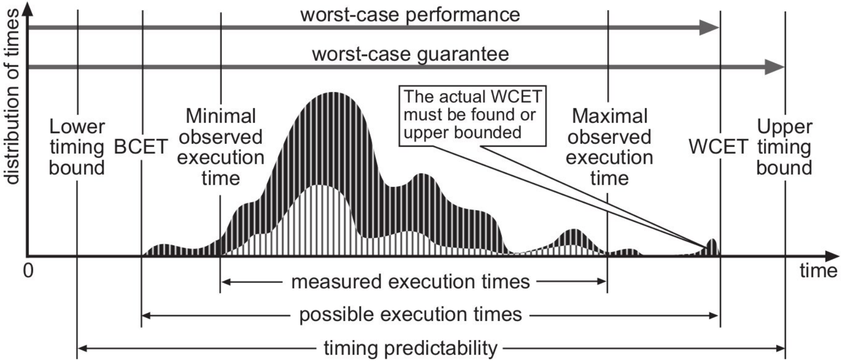
Modern processors: the time it takes to execute the same instruction varies, depending on the history of computations.

2 HW Models for SW People: Timing

- Time Complexity vs Execution Time
- "Real-Time" and the 3 Categories of Computer Systems
- Execution Time: What's in the HW, What's in the SW?
- The Notion of Worst-Case Execution Time (WCET)
- Measuring (Physical) Time as a Security Attack
- HW Models for WCET Estimation: The Cache Problem

Distribution of Actual Execution Times

Same Program, Many Data (and Many Execution Conditions)



From: “The worst-case execution-time problem — overview of methods and survey of tools”. ACM Transactions on Embedded Computing Systems (TECS) Volume 7 Issue 3, April 2008 Article No. 36

Actual Execution Time depends on...

- Details on the execution platform (the hardware, the OS, the network, ...)
- Even for a single isolated machine with an old-style processor (e.g., a good old Motorola 68000 — no pipeline, no cache, no speculation): the execution time of a program depends on its flow of control, that cannot be known statically.

2 HW Models for SW People: Timing

- Time Complexity vs Execution Time
- “Real-Time” and the 3 Categories of Computer Systems
- Execution Time: What’s in the HW, What’s in the SW?
- The Notion of Worst-Case Execution Time (WCET)
- Measuring (Physical) Time as a Security Attack
- HW Models for WCET Estimation: The Cache Problem

Meltdown/Spectre and Side-Channel Attacks

Code snippet from: `blog.cyberus-technology.de/posts/2018-01-03-meltdown.html`

See also *http://www-verimag.imag.fr/~maraninx/research/meltdown/*

```
; rcx = a protected kernel memory address
; rbx = address of a large array in user space
mov al, byte [rcx]          ; read from forbidden kernel address
shl rax, 0xc                ; multiply the result from
                            ; the read  operation with 4096
mov rbx, qword [rbx + rax]  ; touch the user space array at
                            ; the offset that we just calculated
```

First `mov` should be forbidden; but speculative execution starts to execute the `mov` and the two other instructions before the check finally returns; then the functional effet of the instructions are cancelled. But not the indirect effects on the fact that memory locations are in the cache. Start measuring the time it takes to access all the cells in the user array. If the timing is short for a cell, its offset reveals the forbidden byte.

2 HW Models for SW People: Timing

- Time Complexity vs Execution Time
- "Real-Time" and the 3 Categories of Computer Systems
- Execution Time: What's in the HW, What's in the SW?
- The Notion of Worst-Case Execution Time (WCET)
- Measuring (Physical) Time as a Security Attack
- HW Models for WCET Estimation: The Cache Problem

The (WC)ET is Due to: SW Structure + HW Behaviour

- SW side: `for (i=0;i<10;i++) { A(i); }` is probably longer than `for (i=0;i<5;i++) { A(i); }`
- HW side: in the sequence
  `mov (%r0), %r1 ; mov (%r0), %r1`
  (where `(%r0)` is a memory access) the second `mov` can be significantly shorter than the first one (if the first `mov` has put the word in the cache).

Static vs Dynamic Evaluation

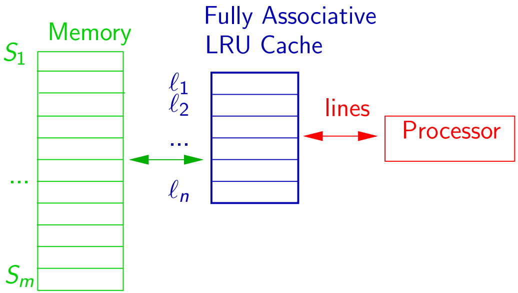
- Dynamic Evaluation: just testing, on the real HW, with usual coverage problems
  We can never know whether there exists a non-explored case that exhibits a greater execution time than those observed.
  For `void spend_time_with_WCET (..., 1)` the execution path is always the same.
  The variation comes from the HW+OS.

- Static Evaluation: a kind of static program analysis, needs a *model* of the HW+OS behaviour, specific techniques with ILP (Integer Linear Programming) for the SW various execution paths.
  Can give rough but guaranteed bounds.

2 HW Models for SW People: Timing

- Time Complexity vs Execution Time
- “Real-Time” and the 3 Categories of Computer Systems
- Execution Time: What’s in the HW, What’s in the SW?
- The Notion of Worst-Case Execution Time (WCET)
- Measuring (Physical) Time as a Security Attack
- HW Models for WCET Estimation: The Cache Problem
  - Behavior of a Fully-Associative LRU Cache
  - Analyzing the SW

Definition of the cache



When the cache is full, and the processor needs a line of the memory that is not in the cache, one line has to be evicted. LRU = the least recently used is evicted and replaced by the new one.

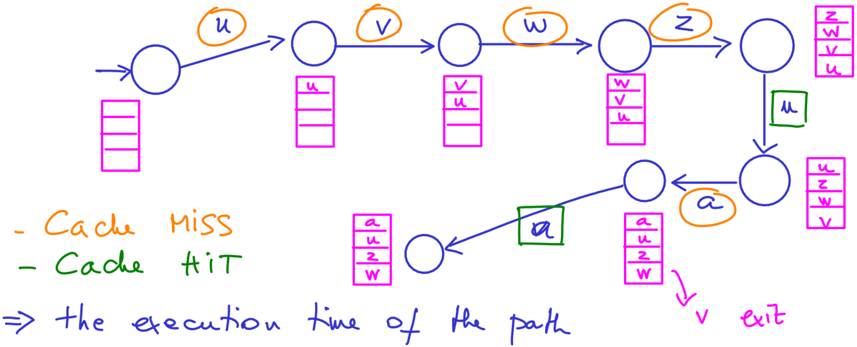
2 HW Models for SW People: Timing

- Time Complexity vs Execution Time
- "Real-Time" and the 3 Categories of Computer Systems
- Execution Time: What's in the HW, What's in the SW?
- The Notion of Worst-Case Execution Time (WCET)
- Measuring (Physical) Time as a Security Attack
- HW Models for WCET Estimation: The Cache Problem
  - Behavior of a Fully-Associative LRU Cache
  - Analyzing the SW

Detailed Control Flow Graph with Memory Accesses

Each transition is labeled by a set of lines, corresponding to the data that are accessed by the operations of the transition.

Example: for a transition `x > 3` (resp. `y++`) in the original detailed control flow graph, we build a transition $\{l_x\}$ (resp. $\{l_y\}$) in the graph to be analysed.
l_x (resp. l_y) is the line in the memory where variable `x` (resp, `y`) has been installed.



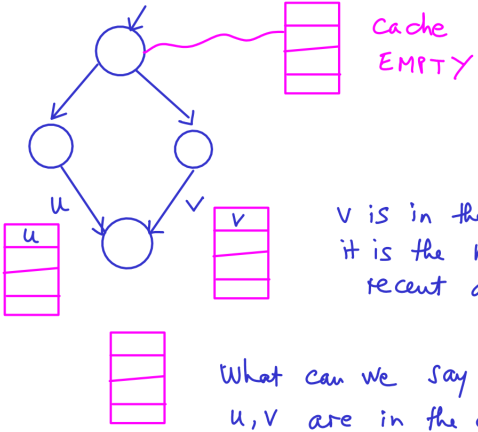
⇒ the execution time of the path is

$$\underbrace{5 \times \text{MISS} + 2 \ \text{HIT}}_{\text{better estimation (with a model of the cache)}} \leq \underbrace{7 \times \text{worst-case access-time (MISS)}}_{\text{if NO INFO on the behavior of the cache}}$$

Need for Abstractions: Example

```
if x {
  access u
} else {
  access v
}
```

Assume the cache has 4 lines. What do we know on the contents of the cache at the end of the IF statement? Is the information exact?



Same for u

v is in the cache
it is the most
recent access

What can we say here ?
u, v are in the cache

but how long will they
remain in the cache
(in the worst case) ?

define their
worst-case age

Conclusion

- The actual execution time depends on the state of the HW (here, the cache). Hence it depends on the history of the execution.
- In dynamic evaluations, the history is unique, but in static evaluations, there are several histories leading to the same "state" in the control flow of the software
- Static evaluation of the WCET requires a model of the cache + *abstractions*[4]

[4] https://en.wikipedia.org/wiki/Abstract_interpretation

1. The Big Picture: from Physics to Software
2. HW Models for SW People: Timing
3. HW Models for SW People: Energy Consumption and Temperature
   - Complete Simulation Models
   - Other Topics of Interest

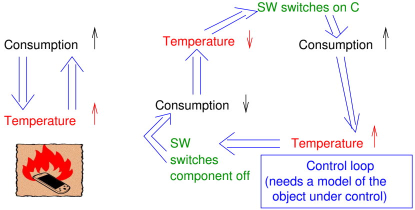
3 HW Models for SW People: Energy Consumption and Temperature

- Complete Simulation Models
- Other Topics of Interest

Main Ideas

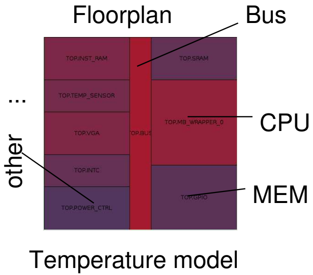
Capture all potential interactions between: voltage, frequency, consumption, temperature, software decisions, state of the battery, ...

Recall:



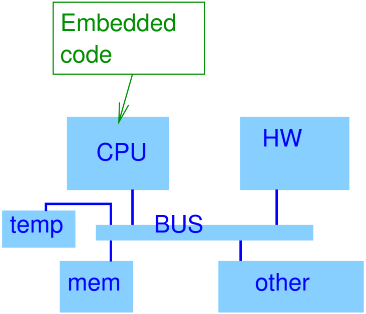
In the most detailed models one can play the actual software on top of a functional+extra-functional model of the hardware.

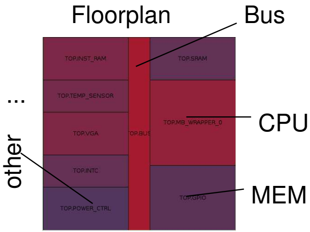
Precise Simulation Models with Temperature Models and Actual Embedded Code



Precise Simulation Models with Temperature Models and Actual Embedded Code

```
if (T > thr) { switch "other" off; turn CPU to (V1,F1) }
```

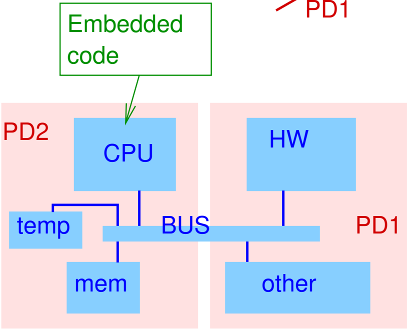




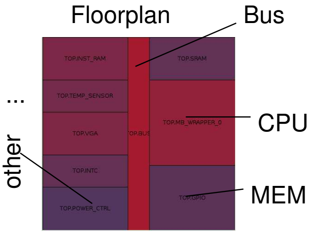
Temperature model

Precise Simulation Models with Temperature Models and Actual Embedded Code

if (T > thr) { switch "~~other~~" off; turn CPU to (V1,F1) } PD1



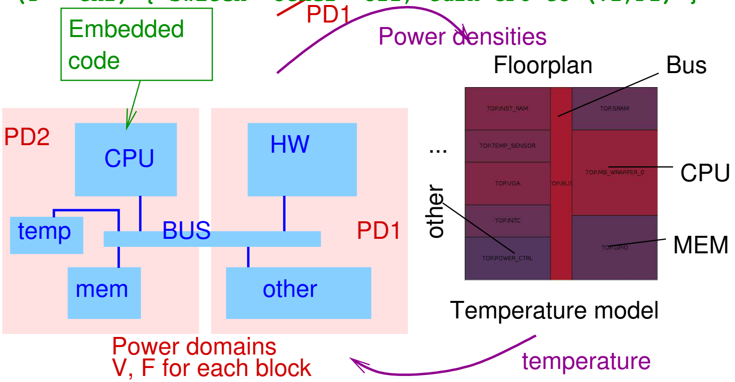
Power domains
V, F for each block



Temperature model

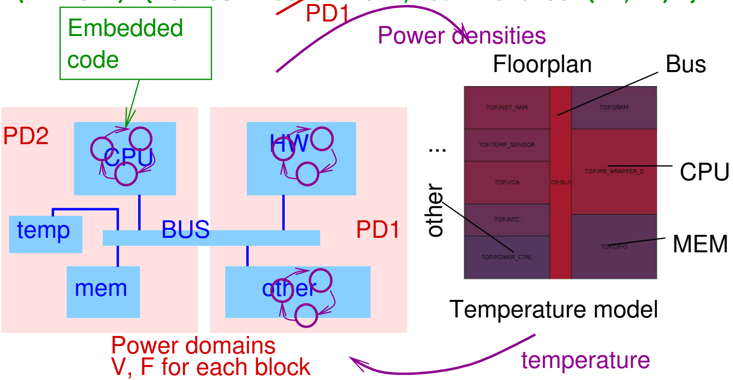
Precise Simulation Models with Temperature Models and Actual Embedded Code

if (T > thr) { switch "~~other~~" off; turn CPU to (V1,F1) }



Precise Simulation Models with Temperature Models and Actual Embedded Code

```
if (T > thr) { switch "~~other~~" off; turn CPU to (V1,F1) }
```



PD1

Power densities

Embedded code

Floorplan

Bus

PD2

CPU

HW

temp

BUS

PD1

mem

other

...

other

CPU

MEM

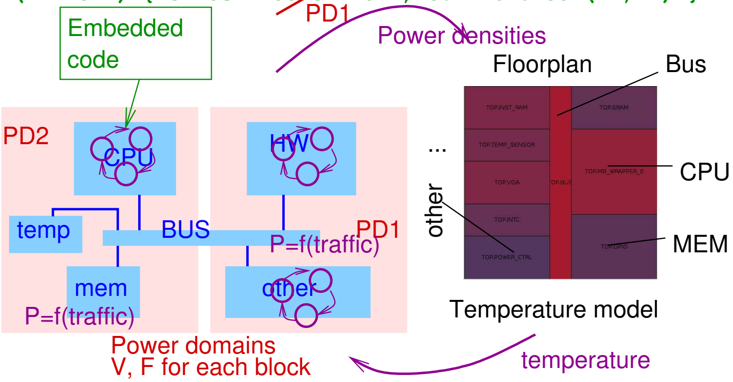
Temperature model

Power domains
V, F for each block

temperature

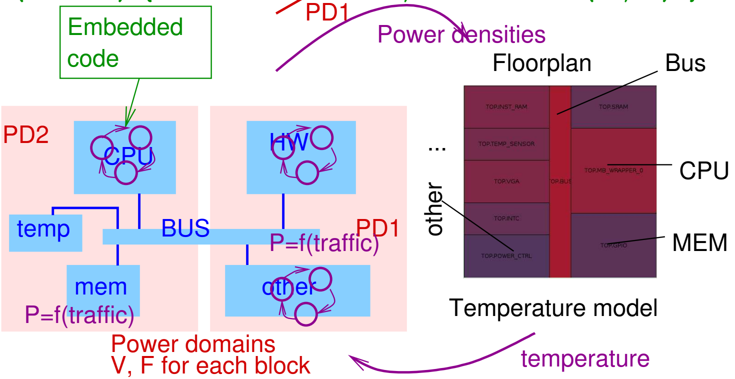
Precise Simulation Models with Temperature Models and Actual Embedded Code

if (T > thr) { switch "~~other~~" off; turn CPU to (V1,F1) }



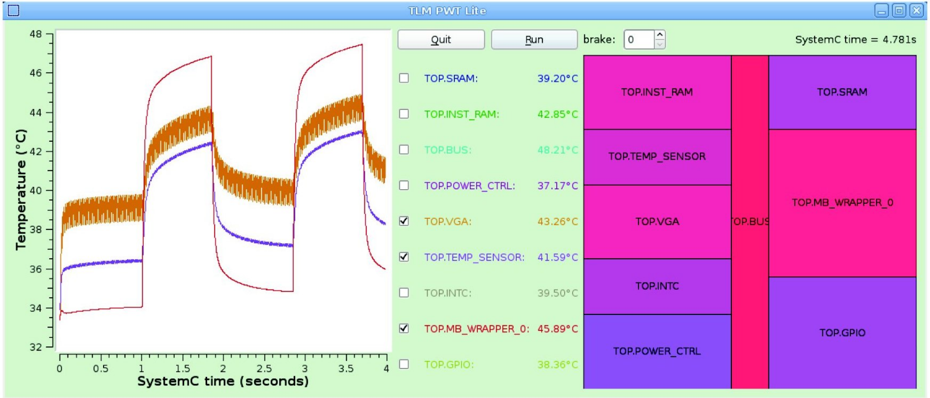
Precise Simulation Models with Temperature Models and Actual Embedded Code

if (T > thr) { switch "~~other~~" off; turn CPU to (V1,F1) }



P=f(traffic) may take contention into account; the Joule–per–bit model cannot.

Example Simulation Results



brake: 0

SystemC time = 4.781s

□ TOP.SRAM: 39.20°C

□ TOP.INST_RAM: 42.85°C

□ TOP.BUS: 48.21°C

□ TOP.POWER_CTRL: 37.17°C

☑ TOP.VGA: 43.26°C

☑ TOP.TEMP_SENSOR: 41.59°C

□ TOP.INTC: 39.50°C

☑ TOP.MB_WRAPPER_0: 45.89°C

□ TOP.GPIO: 38.36°C

Modeling Power Consumption and Temperature in TLM Models — Matthieu Moy, Claude Helmstetter, Tayeb Bouhadiba, Florence Maraninchi - Leibnitz Trans. on Emb. Syst. 2016

A Major Problem: Validation of the Models

A precise simulation is theoretically feasible (at gate level, or even below). But it's terribly slow.

Raising the level of abstraction to get reasonable-time SW-in-the-loop simulations implies accepting relative results only.

- Simulation, say 5%-precise w.r.t. real system: hopeless
- One objective can be to identify peaks, or the points that trigger the control policy in the SW.

+ what's the sensitivity of the overall model to small variations on the figures attached to states?

3 HW Models for SW People: Energy Consumption and Temperature
- Complete Simulation Models
- Other Topics of Interest

3 HW Models for SW People: Energy Consumption and Temperature

- Complete Simulation Models
- Other Topics of Interest
  - RAPL
  - Energy Bugs
  - Simulation Models and the Quest for “Good” Models

Estimated or Actual Values[5]

"Intel provides RAPL to provide estimated (and on some systems, actual) power measurements that the user can access. But how good are these measurements really? We set out to instrument some machines to find out. We looked primarily at the DRAM RAPL readings but also report a bit on the RAPL CPU and RAPL GPU measurements. "

[5] https://web.eece.maine.edu/~vweaver/projects/rapl/rapl_validation.html

Estimated or Actual Values?[6]

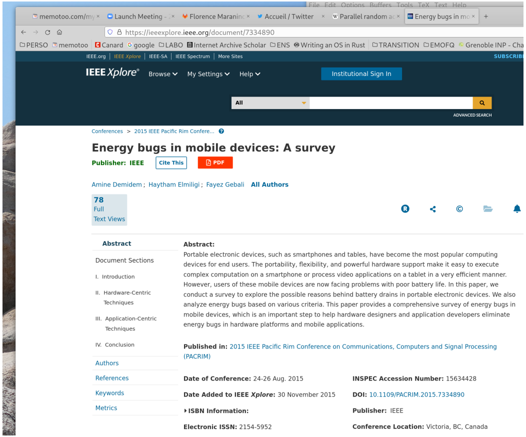
“*In this work, we present a comprehensive study comparing the accuracy of state-of-the-art on-chip power sensors and energy predictive models against system-level physical measurements using external power meters, which we consider to be the ground truth. The measurements provided by on-chip sensors are obtained programmatically using RAPL for Intel multicore CPUs, NVML for Nvidia GPUs and Intel System Management Controller chip (SMC) for Intel Xeon Phis. To compare the approaches reliably, we presented a methodology to determine the component-level dynamic energy consumption of an application using system-level physical measurements using power meters, which are obtained using HCLWattsUp API. (...) We demonstrated through a parallel matrix-matrix multiplication on two Intel multicore CPU servers that using inaccurate energy measurements provided by on-chip sensors for dynamic energy optimization can result in significant energy losses up to 84%.*”

[6] https://mdpi-res.com/d_attachment/energies/energies-12-02204/article_deploy/energies-12-02204.pdf

3 HW Models for SW People: Energy Consumption and Temperature

- Complete Simulation Models
- Other Topics of Interest
  - RAPL
  - Energy Bugs
  - Simulation Models and the Quest for “Good” Models

Energy Bugs: Definition



Energy bugs in mobile devices: A survey

Publisher: IEEE

Amine Demidem ; Haytham Elmiligi ; Fayez Gebali **All Authors**

Abstract:
Portable electronic devices, such as smartphones and tables, have become the most popular computing devices for end users. The portability, flexibility, and powerful hardware support make it easy to execute complex computation on a smartphone or process video applications on a tablet in a very efficient manner. However, users of these mobile devices are now facing problems with poor battery life. In this paper, we conduct a survey to explore the possible reasons behind battery drains in portable electronic devices. We also analyze energy bugs based on various criteria. This paper provides a comprehensive survey of energy bugs in mobile devices, which is an important step to help hardware designers and application developers eliminate energy bugs in hardware platforms and mobile applications.

Published in: 2015 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM)

Date of Conference: 24-26 Aug. 2015

Date Added to IEEE *Xplore*: 30 November 2015

▸**ISBN Information:**

Electronic ISSN: 2154-5952

INSPEC Accession Number: 15634428

DOI: 10.1109/PACRIM.2015.7334890

Publisher: IEEE

Conference Location: Victoria, BC, Canada

Aggressive optimizations for power consumption introduce “new” sources of bugs (aka “energy bugs”, e.g., lock/unlock problems, battery drain but also functional problems)

Characterizing and Detecting Energy Bugs

Wakelocks and other stories, a few references:

- Unit Testing of Energy Consumption of Software Libraries[7]
- Detecting Energy Bugs in Android Apps Using Static Analysis[8]
- Categorization and Detection of Energy Bugs and Application Tail Energy Bugs in Smartphones[9]
- ...

[7] https://hal.archives-ouvertes.fr/hal-00912613

[8] https://link.springer.com/chapter/10.1007/978-3-319-68690_12

[9] https://uwspace.uwaterloo.ca/bitstream/handle/10012/10862/abbasi_abdul.pdf

3 HW Models for SW People: Energy Consumption and Temperature

- Complete Simulation Models
- Other Topics of Interest
  - RAPL
  - Energy Bugs
  - Simulation Models and the Quest for “Good” Models

Models are Necessarily Abstract (see J.-L. Borges)

...In that Empire, the craft of Cartography attained such Perfection that the Map of a Single province covered the space of an entire City, and the Map of the Empire itself an entire Province. In the course of Time, these Extensive maps were found somehow wanting, and so the College of Cartographers evolved a Map of the Empire that was of the same Scale as the Empire and that coincided with it point for point. Less attentive to the Study of Cartography, succeeding Generations came to judge a map of such Magnitude cumbersome, and, not without Irreverence, they abandoned it to the Rigours of sun and Rain. In the western Deserts, tattered Fragments of the Map are still to be found, Sheltering an occasional Beast or beggar; in the whole Nation, no other relic is left of the Discipline of Geography.

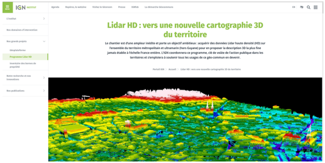
Models are Necessarily Abstract (see J.-L. Borges)

"Del rigor en la ciencia", Jorge Luis Borges[10]
En aquel Imperio, el Arte de la Cartografía logró tal Perfección que el Mapa de una sola Provincia ocupaba toda una Ciudad, y el Mapa del Imperio, toda una Provincia. Con el tiempo, estos Mapas Desmesurados no satisficieron y los Colegios de Cartógrafos levantaron un Mapa del Imperio, que tenía el Tamaño del Imperio y coincidía puntualmente con él. Menos Adictas al Estudio de la Cartografía, las Generaciones Siguientes entendieron que ese dilatado Mapa era Inútil y no sin Impiedad lo entregaron a las Inclemencias del Sol y los Inviernos. En los Desiertos del Oeste perduran despedazadas Ruinas del Mapa, habitadas por Animales y por Mendigos; en todo el País no hay otra reliquia de las Disciplinas Geográficas.

Suárez Miranda: Viajes de varones prudentes,
libro cuarto, cap. XLV, Lérida, 1658.

[10] https://www.madrimasd.org/cienciaysociedad/poemas/poesia.asp?id=247

Lidar HD[11]

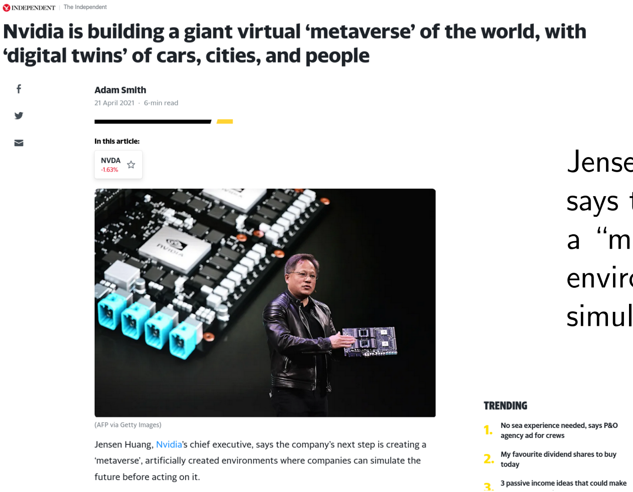


Lidar HD : vers une nouvelle cartographie 3D du territoire

Le chantier est d'une ampleur inédite et porte un objectif ambitieux : acquérir des données Lidar haute densité (HD) sur l'ensemble du territoire métropolitain et ultramarin (hors Guyane) pour en proposer la description 3D la plus fine jamais établie à l'échelle France entière. L'IGN coordonnera ce programme, clé de voûte de l'action publique dans les territoires et s'emploiera à soutenir tous les usages de ce géo-commun en devenir.

[11] https://www.ign.fr/institut/lidar-hd-vers-une-nouvelle-cartographie-3d-du-territoire

... “digital twins” of cars, cities, and people[12]



INDEPENDENT | The Independent

Nvidia is building a giant virtual ‘metaverse’ of the world, with ‘digital twins’ of cars, cities, and people

Adam Smith
21 April 2021 · 6-min read

In this article:

NVDA -1.63% ☆

(AFP via Getty Images)

Jensen Huang, Nvidia’s chief executive, says the company’s next step is creating a ‘metaverse’, artificially created environments where companies can simulate the future before acting on it.

TRENDING

1. No sea experience needed, says P&O agency ad for crews
2. My favourite dividend shares to buy today
3. 3 passive income ideas that could make

Jensen Huang, Nvidia’s chief executive, says the company’s next step is creating a “metaverse”, artificially created environments where companies can simulate the future before acting on it.

[12] https://uk.finance.yahoo.com/news/nvidia-building-giant-virtual-metaverse-115216046.html

Part II

Anti-Limits in Digital Systems

4 A Tale of 3 Futures

5 Questions

6 Anti-Limits

7 This is (Definitely) Not a Conclusion

Discussion based on...

Let Us Not Put All Our Eggs in One Basket:
Towards New Research Directions in Computer Science
F. Maraninchi

To appear in: Communications of the ACM
https://cacm.acm.org/

4 A Tale of 3 Futures

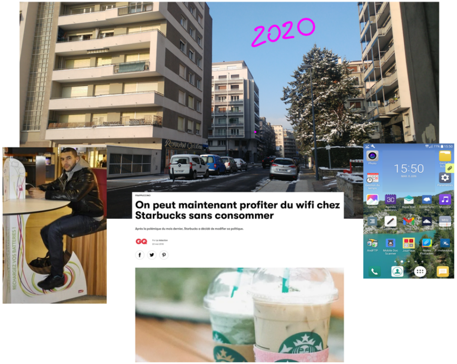
5 Questions

6 Anti-Limits

7 This is (Definitely) Not a Conclusion

Typical Situation in 2005

Typical Situation in 2020



FRAPPUCCINO

On peut maintenant profiter du wifi chez Starbucks sans consommer

Après la polémique du mois dernier, Starbucks a décidé de modifier sa politique.

GQ Par La rédaction
22 mai 2018

Mobile Communications 2005 - 2020

- 2005: Use them to place and receive calls “everywhere”; charge once a week; telephone booths remain; around 140 submarine communications cables[13]: each capable of carrying 3,2 Tbits/s

[13] See https://cablemap.info/_default.aspxorhttps://en.wikipedia.org/wiki/Submarine_communications_cable

Mobile Communications 2005 - 2020

- 2005: Use them to place and receive calls "everywhere"; charge once a week; telephone booths remain; around 140 submarine communications cables[13]: each capable of carrying 3,2 Tbits/s
- 2005 ... 2020: Huge improvements of the devices (hardware, software, batteries, screens, casing, ...) + huge improvements of the infrastructure

[13] See https://cablemap.info/_default.aspxorhttps://en.wikipedia.org/wiki/Submarine_communications_cable

Mobile Communications 2005 - 2020

- 2005: Use them to place and receive calls “everywhere”; charge once a week; telephone booths remain; around 140 submarine communications cables[13]: each capable of carrying 3,2 Tbits/s
- 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, ...) there are now around 400 cables, each capable of carrying: 320 Tbit/s

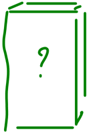
[13] See https://cablemap.info/_default.aspxorhttps://en.wikipedia.org/wiki/Submarine_communications_cable

Theoretically Thinkable Paths

Charge once a week

HUGE improvements in batteries + HW, SW, ...

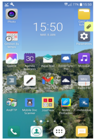
- Same functions charge once a month (?)



Charge once a week more functions

Direct-filled solar charger
The sunlight directly charges the electronic device. It can also charge mobile power.

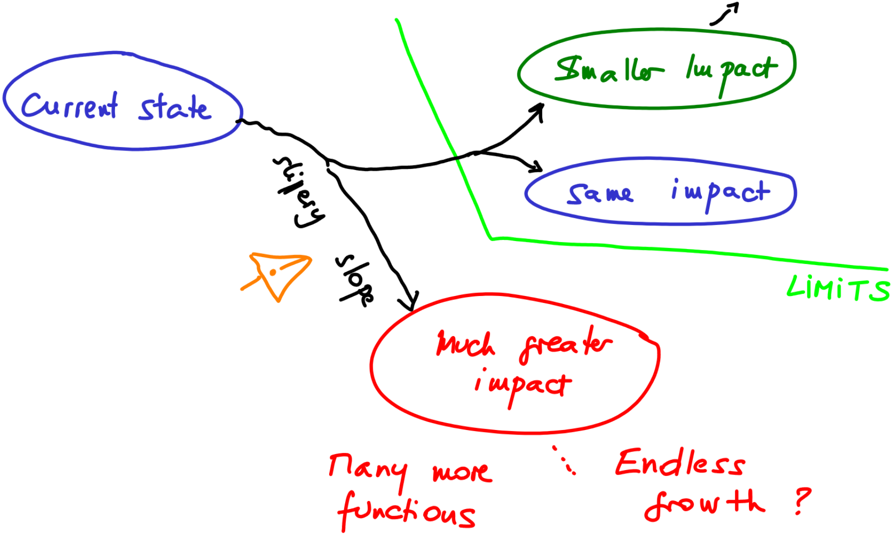
+



Charge twice-a-day

Many more functions

Several Paths...



4 A Tale of 3 Futures

5 **Questions**

6 Anti-Limits

7 This is (Definitely) Not a Conclusion

How to Avoid Future 3 (The Slippery Slope)? Optimizations vs Rebound Effects

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, from various points of view: technological, economical, sociological, etc., in order to try and reproduce it. If there is no such example, then we should stop believing that optimizations always help reducing environmental impacts.

Start thinking in terms of limits.
First step: identify anti-limits.

4 A Tale of 3 Futures

5 Questions

6 Anti-Limits

7 This is (Definitely) Not a Conclusion

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)[14], PKT (proof of bandwidth)[15], NFTs, etc.)

[14] https://en.wikipedia.org/wiki/Chia_(cryptocurrency)

[15] 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)[14], PKT (proof of bandwidth)[15], NFTs, etc.)
- Provisions resources for immediate service delivery, whatever the number of clients

[14] https://en.wikipedia.org/wiki/Chia_(cryptocurrency)

[15] 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)[14], PKT (proof of bandwidth)[15], NFTs, etc.)
- Provisions resources for immediate service delivery, whatever the number of clients
- Assumes unlimited storage — in space and time (initial advertisement of gmail)

[14] https://en.wikipedia.org/wiki/Chia_(cryptocurrency)

[15] 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)[14], PKT (proof of bandwidth)[15], 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)

[14] https://en.wikipedia.org/wiki/Chia_(cryptocurrency)

[15] 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)[14], PKT (proof of bandwidth)[15], 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) × (growing number of users)

[14] https://en.wikipedia.org/wiki/Chia_(cryptocurrency)

[15] 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)[14], PKT (proof of bandwidth)[15], 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) × (growing number of users)
- Built to allow for "unlimited" functional extensions (web)

[14] https://en.wikipedia.org/wiki/Chia_(cryptocurrency)

[15] 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)[14], PKT (proof of bandwidth)[15], 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) × (growing number of users)
- Built to allow for "unlimited" functional extensions (web)
- Bets on the availability of a better/bigger/more efficient machine, next year (SW obesity)

[14] https://en.wikipedia.org/wiki/Chia_(cryptocurrency)

[15] 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)[14], PKT (proof of bandwidth)[15], 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) × (growing number of users)
- Built to allow for "unlimited" functional extensions (web)
- Bets on the availability of a better/bigger/more efficient machine, next year (SW obesity)
- Deployment is profitable only if there are more users and/or increasing usage per user (5G)

[14] https://en.wikipedia.org/wiki/Chia_(cryptocurrency)

[15] https://pkt.cash/

Candidate Limits (and Induced Constraints)

- Gemini (heavier than gopher, lighter than the web, ...)[16] - same idea as restricted DSLs, no images, no extensions, ...

[16] https://gemini.circumlunar.space/

[17] https://collapseos.org/

Candidate Limits (and Induced Constraints)

- Gemini (heavier than gopher, lighter than the web, ...)[16] - same idea as restricted DSLs, no images, no extensions, ...
- All configurations in:
  $\{$ intermittent, quotas $\} \times \{$ power, connection, memory, computing power $\}$

[16] https://gemini.circumlunar.space/

[17] https://collapseos.org/

Candidate Limits (and Induced Constraints)

- Gemini (heavier than gopher, lighter than the web, ...)[16] - same idea as restricted DSLs, 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

[16] https://gemini.circumlunar.space/

[17] https://collapseos.org/

Candidate Limits (and Induced Constraints)

- Gemini (heavier than gopher, lighter than the web, ...)[16] - same idea as restricted DSLs, 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

[16] https://gemini.circumlunar.space/

[17] https://collapseos.org/

Candidate Limits (and Induced Constraints)

- Gemini (heavier than gopher, lighter than the web, ...)[16] - same idea as restricted DSLs, 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?

[16] https://gemini.circumlunar.space/

[17] https://collapseos.org/

Candidate Limits (and Induced Constraints)

- Gemini (heavier than gopher, lighter than the web, ...)[16] - same idea as restricted DSLs, 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*[17]

[16] `https://gemini.circumlunar.space/`

[17] `https://collapseos.org/`

An Example Question: is Extensibility a Desirable Property?

> *Extensibility[18] 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?

[18] https://en.wikipedia.org/wiki/Extensibility

4. A Tale of 3 Futures

5. Questions

6. Anti-Limits

7. This is (Definitely) Not a Conclusion

Limits as First-Class Citizen

Static, beforehand evaluation of Worst-Case-Resource-Consumption as part of the requirements.