Welcome to the first virtual SoftAWERE Workshop

Creating the tools necessary to measure the energy efficiency of open source libraries and components, creating transparency and enabling better choices for developers

We would like to introduce ourselves

A warm welcome from your project team

This workshop aims to enable you to contribute with your expertise

We will explain why the project exists, what we are trying to solve, and how we are planning to solve it.

You will get the chance to discuss the topics of your choice in the breakout rooms and help us to move this project forward.

In the end, we will consolidate and filter the main discussion points in the plenary to derive specific actions from your input.

Time	Agenda Item
09:45-10:00 (optional)	**Introduction to the Miro Tool** Optional introduction to Miro, which will be used for the interactive part of the workshop
10:00-10:15	**Welcome and Introductions** • Greetings & Setting the stage from UBA and BMWK • Kick-Off SDIA • Workshop Format
10:15-10:45	**Presentation of the Project**
10:45-11:20	**Breakout Room Discussions - Round 1**
	Room 1: Methodology: How can energy efficiency of software be measured?
	Room 2: Application: What usage scenarios do we envision and where do we see the highest impact on potential software efficiency improvements?
	Room 3: Outlook: What is the current perception of energy and resource efficiency inside the software community? Where are boundaries towards sustainability?
11:20-11:25	**Change of Rooms** After 35 minutes, we will mix up the rooms again, so everyone can contribute to more than one topic
11:25-12:00	**Breakout Room Discussions - Round 2** Rooms stay the same as above
BREAK	
12:45-13:15	**Discussion of Results**
13:15-13:50	**Open Q&A and Feedback**
13:50-14:00	**Wrap Up and Next Steps**

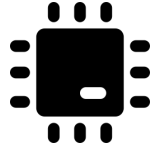
Software consumes resources.

Software also consumes resources.

Network Bandwidth

Storage Capacity

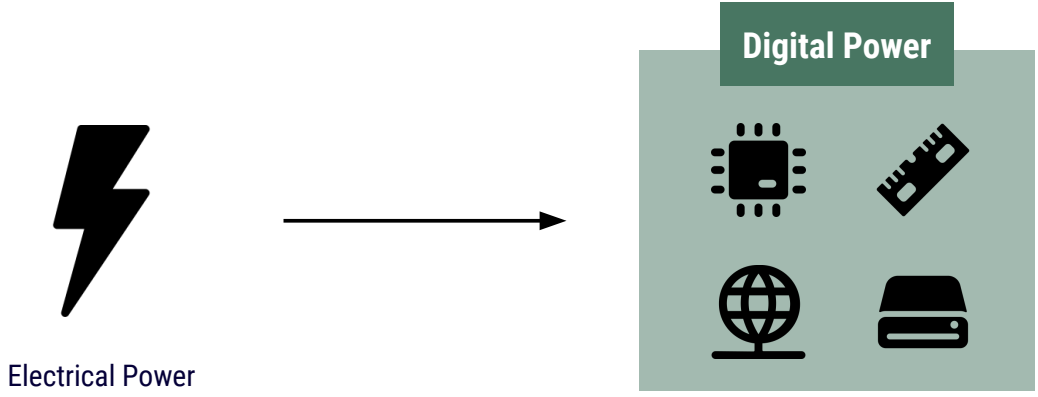
Memory Capacity



CPU Time

Digital Power

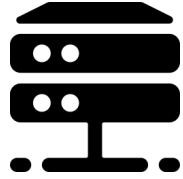
Digital power is created by converting electricity into bits - for processing, storage or transport.



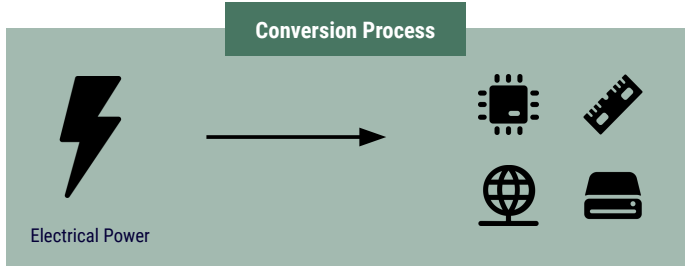
The "conversion machines" can be small-scale or large-scale.

Smartphone

Laptop



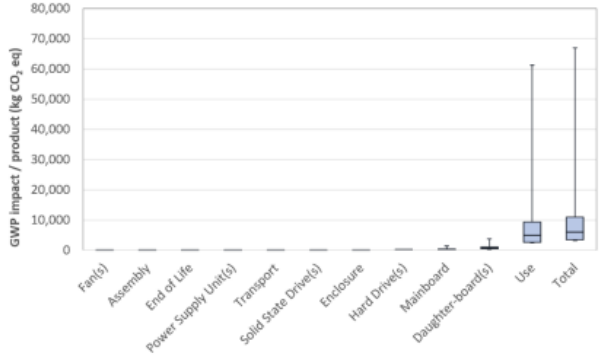
Server



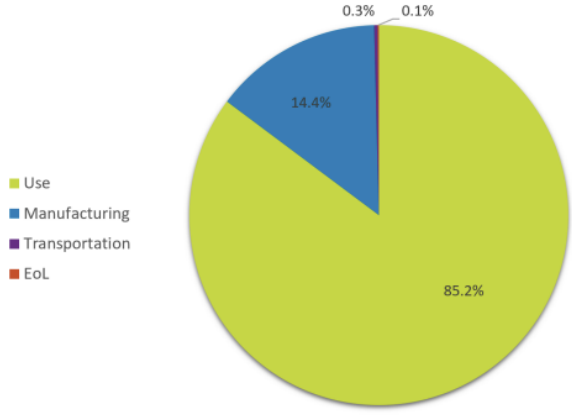
Its conversion machines which are creating the footprint for software - resource and electricity use.

Electricity

Resources



* This product has an estimated standard deviation of +/- 10600 kgCO2e



9180 kgCO2e

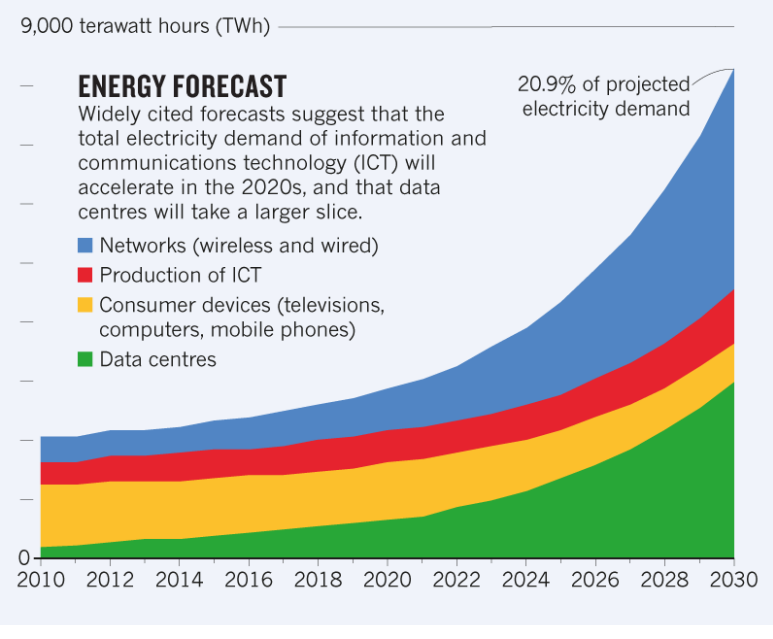
PowerEdge R740XD

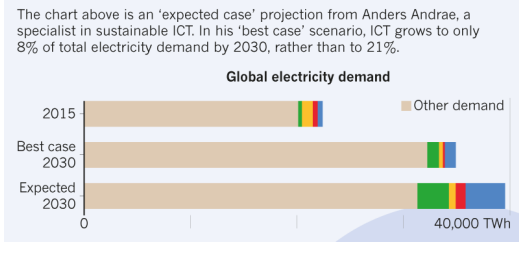
Source: Dell

Its conversion machines which are creating the footprint for software - resource and electricity use.

Electricity

Resources



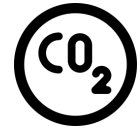


Software has an environmental footprint, based on electricity & resource consumption of the digital power conversion machines

Environmental
Footprint of Software

Electricity

Resources



Embedded CO2-eq

Digital Power

Sustainable Software Design (SSD) Project

Jens Gröger
Senior Researcher for sustainable ICT at Oeko-Institut e.V.

SDIA

Öko-Institut e.V.

Research Project Sustainable Software Design (SSD)

Development and Application of Criteria for resource-efficient Software Products with Consideration of existing Methods

- Commissioned by German Federal Environment Agency (UBA)
- Published: December 2018
- https://www.umweltbundesamt.de/publikationen/entwicklung-anwendung-von-bewertungsgrundlagen-fuer

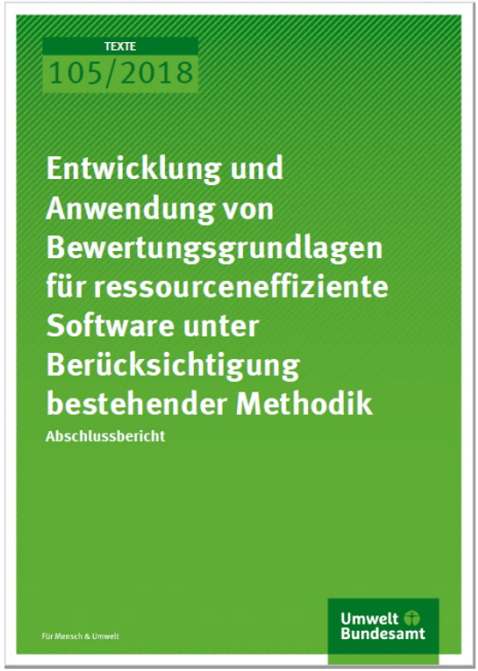
Research group

- Oeko-Institut e.V., Products and Material Flows Division
  - Expertise in the sustainability analysis of ICT devices, Development of award criteria for eco-labels
- University of Zurich, Computer Science and Sustainability
  - Expertise in life cycle assessment of ICT devices and infrastructure
- Trier University of Applied Sciences, Environmental Campus Birkenfeld, Institute for Software Systems in Economy, Environment and Administration
  - Expertise in evaluation and software, Green Software and Green Software Engineering

Öko-Institut e.V.
Institut für angewandte Ökologie
Institute for Applied Ecology

Universität Zürich[UZH]

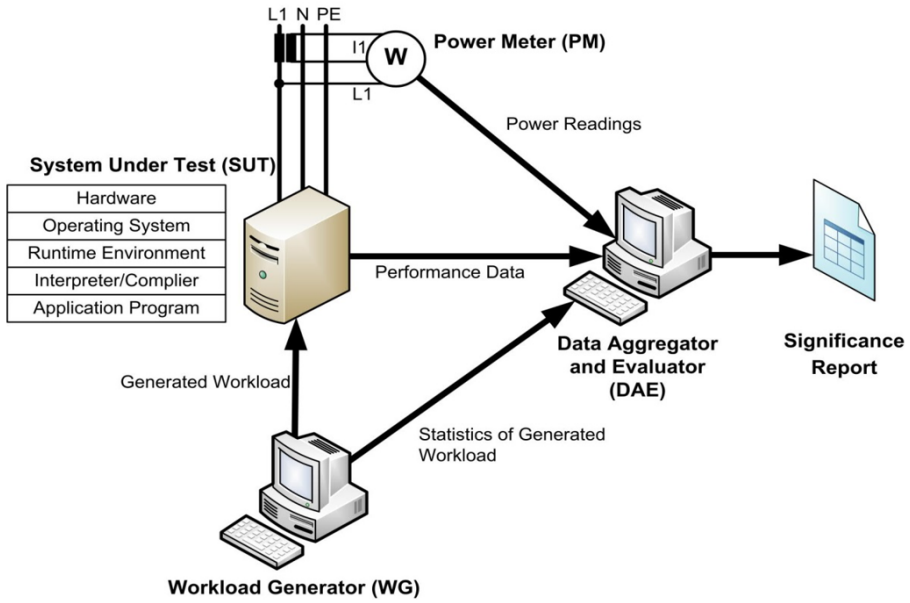
HOCHSCHULE TRIER
Umwelt-Campus Birkenfeld



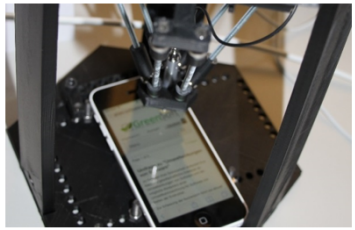
Research Questions

- Is there a **connection** between software and resource consumption?
- Is it possible to define **criteria** for assessing the resource use of software?
- Are such criteria suitable for **comparing** different software products?
- Can **minimum requirements** be defined that describe resource-efficient software?

Measurement Setup

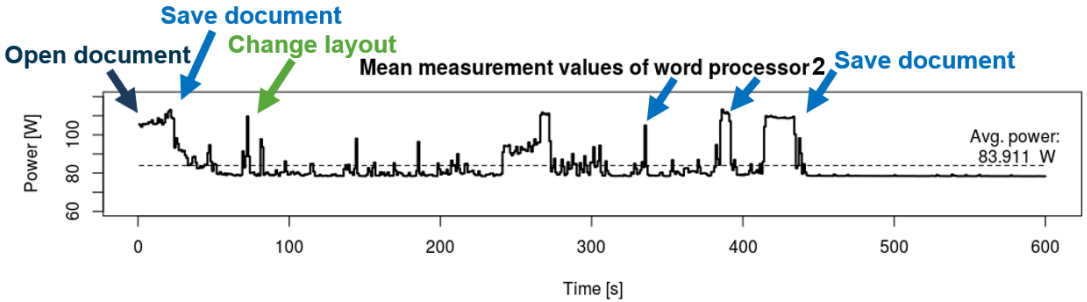


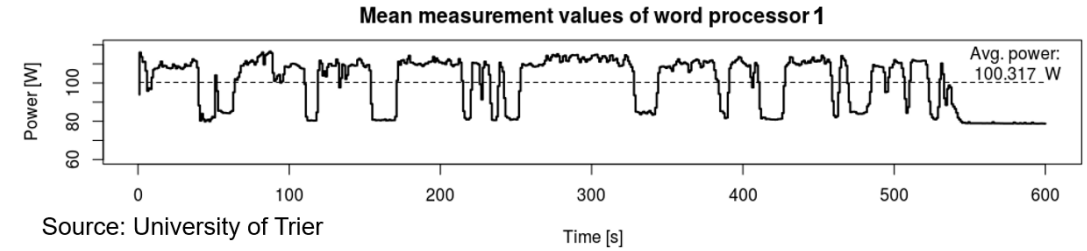
Source: University of Trier



Example of a measurement cycle

- Measurement of **word processing software** usage over 10 minutes



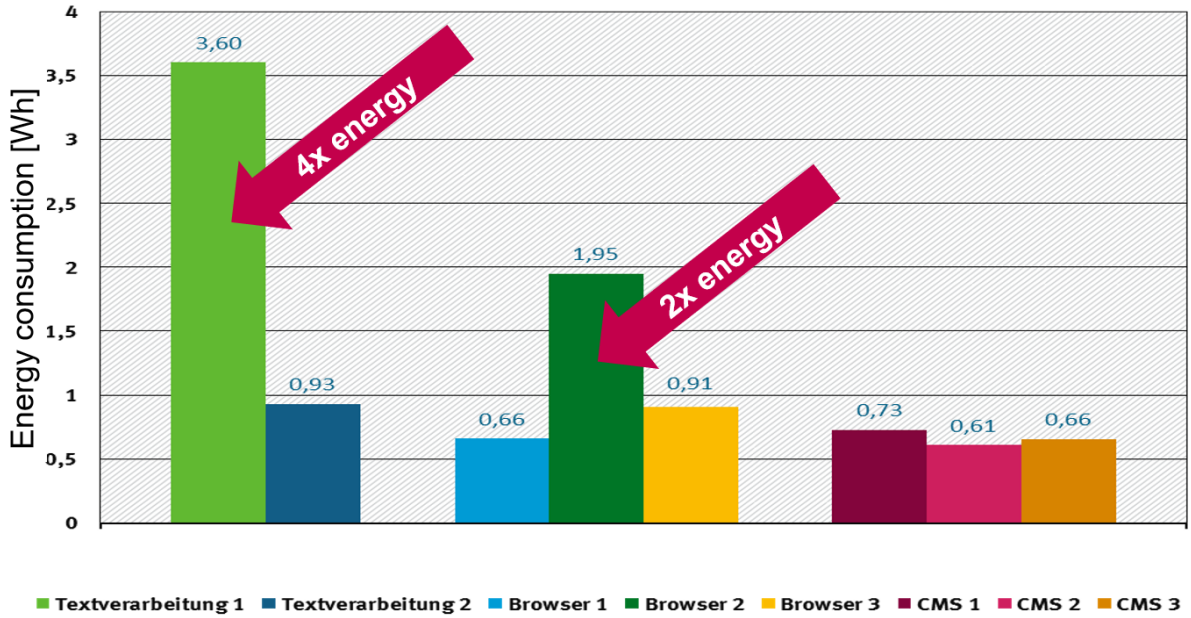


Source: University of Trier

→ Word processor 1 increases average power consumption by 16 watts

Exemplary measurement results usage scenario

- Energy consumption during execution of a standard usage scenario



Most important Results

- Software is responsible for **energy consumption** and **hardware usage** and it can contribute to hardware obsolescence.
- A **set of criteria for sustainable software** was developed which is suitable for the assessment and optimisation of software.
- The application of the criteria catalogue shows **clear differences** between different software products.
- It was possible to **identify criteria** that are suitable for an eco-label for software products.

Enabling IT developers & architects to understand, measure and improve this footprint, therefore is of critical importance to society and businesses addressing their environmental impact.

Majority of software today is build by assembling open source libraries and components.

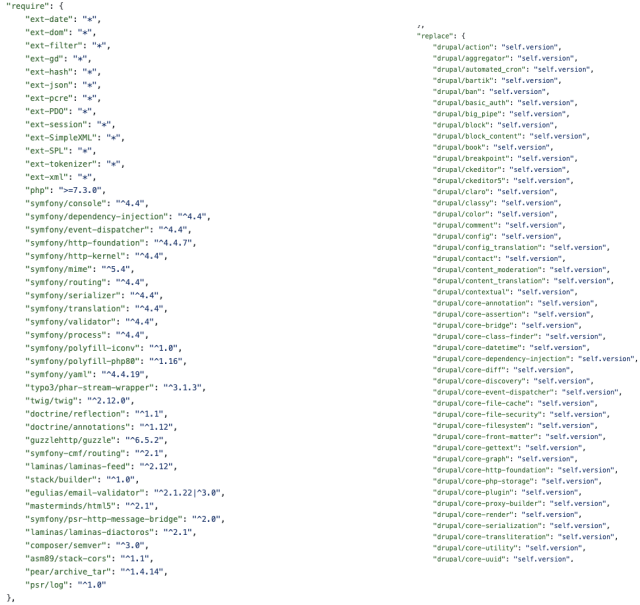
Top ways enterprise open source is being used

1. IT infrastructure modernization **64%**
2. Application development **54%**
3. Digital transformation **53%**

Red Hat

[The State of Enterprise Open Source]

So doesn't the choice of components already impact the electricity footprint of an application?



```
"require": {
    "ext-date": "*",
    "ext-dom": "*",
    "ext-filter": "*",
    "ext-gd": "*",
    "ext-hash": "*",
    "ext-json": "*",
    "ext-pcre": "*",
    "ext-PDO": "*",
    "ext-session": "*",
    "ext-SimpleXML": "*",
    "ext-SPL": "*",
    "ext-tokenizer": "*",
    "ext-xml": "*",
    "php": ">=7.3.0",
    "symfony/console": "^4.4",
    "symfony/dependency-injection": "^4.4",
    "symfony/event-dispatcher": "^4.4",
    "symfony/http-foundation": "^4.4.7",
    "symfony/http-kernel": "^4.4",
    "symfony/mime": "^5.4",
    "symfony/routing": "^4.4",
    "symfony/serializer": "^4.4",
    "symfony/translation": "^4.4",
    "symfony/validator": "^4.4",
    "symfony/process": "^4.4",
    "symfony/polyfill-iconv": "^1.0",
    "symfony/polyfill-php80": "^1.16",
    "symfony/yaml": "^4.4.19",
    "typo3/phar-stream-wrapper": "^3.1.3",
    "twig/twig": "^2.12.0",
    "doctrine/reflection": "^1.1",
    "doctrine/annotations": "^1.12",
    "guzzlehttp/guzzle": "^6.5.2",
    "symfony-cmf/routing": "^2.1",
    "laminas/laminas-feed": "^2.12",
    "stack/builder": "^1.0",
    "egulias/email-validator": "^2.1.22|^3.0",
    "masterminds/html5": "^2.1",
    "symfony/psr-http-message-bridge": "^2.0",
    "laminas/laminas-diactoros": "^2.1",
    "composer/semver": "^3.0",
    "asm89/stack-cors": "^1.1",
    "pear/archive_tar": "^1.4.14",
    "psr/log": "^1.0"
},
,,
"replace": {
    "drupal/action": "self.version",
    "drupal/aggregator": "self.version",
    "drupal/automated_cron": "self.version",
    "drupal/bartik": "self.version",
    "drupal/ban": "self.version",
    "drupal/basic_auth": "self.version",
    "drupal/big_pipe": "self.version",
    "drupal/block": "self.version",
    "drupal/block_content": "self.version",
    "drupal/book": "self.version",
    "drupal/breakpoint": "self.version",
    "drupal/ckeditor": "self.version",
    "drupal/ckeditor5": "self.version",
    "drupal/claro": "self.version",
    "drupal/classy": "self.version",
    "drupal/color": "self.version",
    "drupal/comment": "self.version",
    "drupal/config": "self.version",
    "drupal/config_translation": "self.version",
    "drupal/contact": "self.version",
    "drupal/content_moderation": "self.version",
    "drupal/content_translation": "self.version",
    "drupal/contextual": "self.version",
    "drupal/core-annotation": "self.version",
    "drupal/core-assertion": "self.version",
    "drupal/core-bridge": "self.version",
    "drupal/core-class-finder": "self.version",
    "drupal/core-datetime": "self.version",
    "drupal/core-dependency-injection": "self.version",
    "drupal/core-diff": "self.version",
    "drupal/core-discovery": "self.version",
    "drupal/core-event-dispatcher": "self.version",
    "drupal/core-file-cache": "self.version",
    "drupal/core-file-security": "self.version",
    "drupal/core-filesystem": "self.version",
    "drupal/core-front-matter": "self.version",
    "drupal/core-gettext": "self.version",
    "drupal/core-graph": "self.version",
    "drupal/core-http-foundation": "self.version",
    "drupal/core-php-storage": "self.version",
    "drupal/core-plugin": "self.version",
    "drupal/core-proxy-builder": "self.version",
    "drupal/core-render": "self.version",
    "drupal/core-serialization": "self.version",
    "drupal/core-transliteration": "self.version",
    "drupal/core-utility": "self.version",
    "drupal/core-uuid": "self.version",
```

We believe so. So the first step is to create transparency and enable choice for developers selecting components:

Total **power consumption** during tests:
123 kWh (e)*

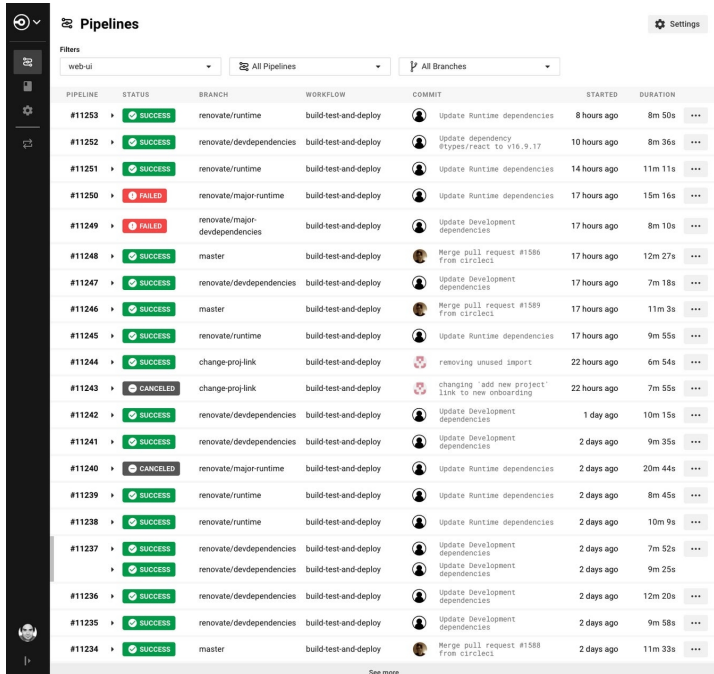
* There is many ways to approach this label = we will evaluate the best approach during the project.

But how to actually measure the energy efficiency of a software library (without applying it)?

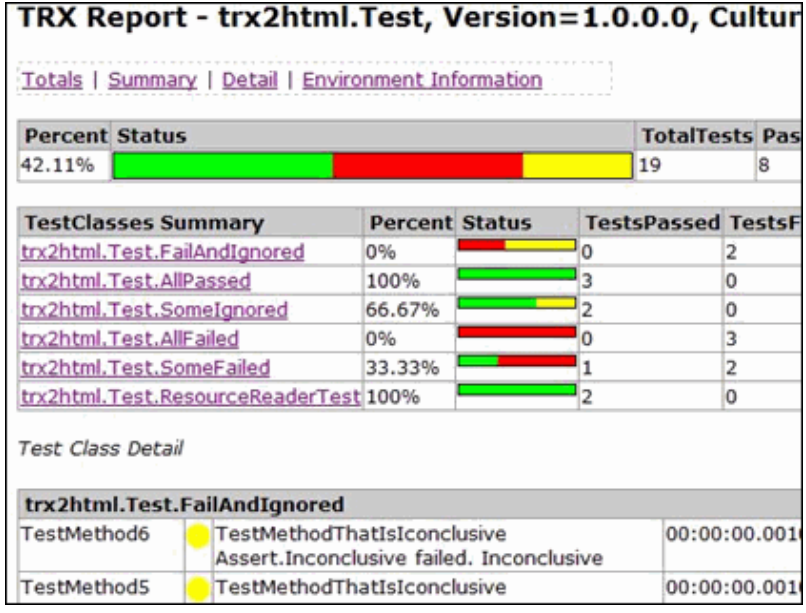
"Software testing is the process of evaluating and verifying that a software product or application does what it is supposed to do." - IBM

" [...] unit testing is a software testing method by which individual units of source code—sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures—are tested to determine whether they are fit for use." - Wikipedia

Scaphandre [skafɑ̃dʁ] is a metrology agent dedicated to electrical power consumption metrics.



Our approach: Record power consumption while each test is executing and report results into CI/CD flow



TRX Report - trx2html.Test, Version=1.0.0.0, Cultur

Totals | Summary | Detail | Environment Information

Percent	Status	TotalTests	Pas
42.11%		19	8

TestClasses Summary	Percent	Status	TestsPassed	TestsF
trx2html.Test.FailAndIgnored	0%		0	2
trx2html.Test.AllPassed	100%		3	0
trx2html.Test.SomeIgnored	66.67%		2	0
trx2html.Test.AllFailed	0%		0	3
trx2html.Test.SomeFailed	33.33%		1	2
trx2html.Test.ResourceReaderTest	100%		2	0

Test Class Detail

trx2html.Test.FailAndIgnored			
TestMethod6		TestMethodThatIsIconclusive Assert.Inconclusive failed. Inconclusive	00:00:00.001
TestMethod5		TestMethodThatIsIconclusive	00:00:00.001

In local development environments, physical power measuring infrastructure won't be available

RAPL vs. Using Constants?

And we will look at your IDEs themselves and how much power they consume when used:

Programming languages:
Python, Java, C, C++, JavaScript, Go Lang

Development environments:
Android Studio, Visual Studio, Eclipse IDE, Jetbrains, Visual Studio Code / GitHub Atom

Last, here is what the project will not be or do:

Benchmarking or comparison of libraries

Have 99.999% accuracy in all scenarios

Consider the complete lifecycle impact of the library

Be applicable for user interfaces & GUI

Let's get started with the Breakout Rooms

SDIA

BREAKOUT ROOMS

	Round 1	Round 2
Room 1: Methodology: How can energy efficiency of software be measured? **Moderator: Jens Gröger**	Johnny Westerlund Fiorenza Oppici Matthias Naab Tanjina Islam Joseph De Vaugh-Geiss Jeroen Burks Rafal Tomczyk Marina Emsing Johannes Brand Peter Fernana-Ritchie	Lutz Veldmann Roberta Haseleu Florian Petri Dennis Gumz Dorian Grosch Christoph Buchli Heiner Von Brachel Elmar Borgmeier Ivano Malavolta
Room 2: Application: What usage scenarios do we envision and where do we see the highest impact on potential software efficiency improvements? **Moderator: Chris Adams**	Lutz Veldmann Benno Schmidt Roberta Haseleu Jutta Eckstein Elmar Borgmeier Marina Köhn Max Schumacher Jing Jing Kris Sharma	Johnny Westerlund Fiorenza Oppici Tanjina Islam Rafal Tomczyk Marina Emsing Andreas Halatsch Johannes Brand Peter Fernana-Ritchie
Room 3: Outlook: What is the current perception of energy and resource efficiency inside the software community? Where are boundaries towards sustainability? **Moderator: Max Schulze**	Florian Petri Dennis Gumz Dorian Grosch Christoph Buchli Heiner Von Brachel Andreas Halatsch Ivano Malavolta	Matthias Naab Joseph De Vaugh-Geiss Jeroen Burks Benno Schmidt Jutta Eckstein Marina Köhn Jing Jing Max Schumacher

SDIA

Welcome back!

SDIA

Let's discuss in the plenary

Each moderator quickly summarizes the main points of the discussion from the respective breakout room (max. 3 minutes)

We will give the larger group the chance to add to the summary (max. 7 minutes per topic)

After the discussion, we will move on with an open Q&A session on general questions that you have after this workshop

ROOM 1

Methodology

How can energy efficiency of software be measured?

Standardization of the interface of resource consumption <-> software

For mobile applications improving the battery lifetime is a regular habit

There is a lack of tools and methodologies to do this

On server side it is easy to measure the energy consumption...

...but it's getting complicated if it comes to the actual effects of the software

Installing Software in containers would be an approach to get an interface to these values

Libraries? Modular design of libraries would be fine and tests, which parts are really used (to leave out the unused parts).

There is an incentive to reduce resource consumption on mini-computers (rasberry pi) and old computers (>10a)

Data transfer (e.g. from mobile applications) can be measured in emulators

Public authorities but also a number of companies are interested in getting external impacts (CO2) from their cloud activities

Providers of cloud services (e.g. usage time of HPC) should also report about energy consumption of their services

SDIA

Application

What usage scenarios do we envision and where do we see the highest impact on potential software efficiency improvements?

right now it's difficult to tell scenarios are high leverage because available data is so poor. Spend one of the few proxies we can use

we only have high level power usage figures , service level are necessary to understand which sectors have the most scope for improvement. ML? Browsers? CMS? Video?

this favours very high scale usage scenarios - without visibility on absolute usage it's difficult to tell where the gains lie

there is a trade-off between visible improvements for intensive users (AI), and smaller, but diffuse improvements across a wide user base (browsers?)

do you need absolute resource budgets to act as an incentive in procurement, with proof you stayed inside them?

what about scenarios outside the library? development vs deployment

scenarios with very tight mapping between usage vs test coverage

Some notion of energy cost per feature used in the wild would be needed to inform which parts of a library need attention.

decoupling is a key issue to track - can we represent this in CI? API compatibility?

Teams with efficiency / sustainability budgets to "buy" efficiency on a project

ROOM 3

Outlook

What is the current perception of energy and resource efficiency inside the software community? Where are boundaries towards sustainability?

Open Q&A

Feel free to ask any additional questions or give feedback on the workshop

Thank you all for joining.

SDIA

ROOM 1

Methodology

How can energy efficiency of software be measured?

1. Do you measure hardware resources when developing software?
2. How do you get access to CPU-Load, Storage usage, network traffic, energy consumption?
3. How do you choose software-languages, libraries, and plug-ins in concern of resource consumption?

Webpage development shifts to smaller frameworks (due to loading time and security)

Measuring power use of one server is easy, virtualized systems are not trivial

CPU time of a server/ calls
dc need to provide access

some dc cash for calculation time

Observation: Objective of discussion: avoiding deployment (containarization, scalability)
Proposal: What measurement should we apply to improve decision making? It doesnt have to be energy measurement.

how much battery is used (in a google glass)?

energy usage (for all mobile devices)
e.g. appmeter

In general people don't decide for a library because of energy usage

gnu systems (eg. debian) for small (raspberrypi) and very old (10years+) devices need to look at power consumption

SDIA

Methodology

How can energy efficiency of software be measured?

in virtualized context, software that is ready for deployment

if started from scratch, the results of this project. what about existing software?

There are programs that track if libraries are used

On the Android platform, as energy drain is a serious issue

trying by doing in the own server room

clean code habits without measuring the impact (feeling for speed of computer)

data transfer amount and carbon footprint of mobile apps (differentiate between core functionality and ad and tracking)

standard usage scenario recording script (man in the middle proxy)

lack of methodology for everyday work

SDIA

Application - round 1 - Max Schumacher

What usage scenarios do we envision and where do we see the highest impact on potential software efficiency improvements?

Good architecture helps not wasting resources (e.g. modularity)

Schedule non-time critical jobs to run at time with high renewable energy mix (in Cloud)

Avoid code waste

ROOM 2

Application - round 1

What usage scenarios do we envision and where do we see the highest impact on potential software efficiency improvements?

5 mins ***by yourself*** try answering the question, and share your rationale when you do

5 mins ***together*** share back

10 mins ***together*** group reflection and discussion

5 mins ***together*** recap, and capture key points as sticky notes for main board

SDIA

Application - round 1 - Benno Schmidt

What usage scenarios do we envision and where do we see the highest impact on potential software efficiency improvements?

where a component is called up very often, there is potential (e.g. if a function is called a million times per second).

sensitize developers to the topic

integrate the idea of sustainabilty into the software development process (including social sustainability

teach students about existing methodologies

Application - round 1 - Roberta Haseleu

What usage scenarios do we envision and where do we see the highest impact on potential software efficiency improvements?

Include the idea of sustainability right from the start of software design

Understand how green coding is different from "just" good coding

Apply a set of best practices as a standard approach in any software project

A way of proving greeness to customers

Application - round 1 - Jutta Eckstein

What usage scenarios do we envision and where do we see the highest impact on potential software efficiency improvements?

Ensuring that software is uncoupled from the hardware

Making it mandatory for all software to make the energy consumption transparent
(and the mandate might come with enough companies just doing that)

Modern architecture, micro services, etc.

Monitor if features are used (and if not get rid of them)

Providing a tool/framework that people can plug in/use when developing individual software

SDIA

ROOM 2

Application - round 2

What usage scenarios do we envision and where do we see the highest impact on potential software efficiency improvements?

5 mins ***by yourself*** try answering the question, and share your rationale when you do

5 mins ***together*** share back

10 mins ***together*** group reflection and discussion

5 mins ***together*** recap, and capture key points as sticky notes for main board

SDIA

Application - round 2 - Tanjina Islam

What usage scenarios do we envision and where do we see the highest impact on potential software efficiency improvements?

when you have limited resource to run the software application on

while trasferring data or communicating through the network

Application - round 2 - Johannes Brand

What usage scenarios do we envision and where do we see the highest impact on potential software efficiency improvements?

I think we should focus on the time, cpu and memory consuming tasks.

Also very important for "off-grid" applications -> smart devices?

It might be substantial to focus on efficencies not in each individuals code but especally in popular libraries.

Application - round 2 - Peter Fernana-Ritchie

What usage scenarios do we envision and where do we see the highest impact on potential software efficiency improvements?

My first thought is "how do we define software efficiency"?

Templated software deployment environment that allows code to be "measured" in a consistent manner

Highest impact for developers that I can see is training - how to develop efficient code given todays deployment options

A framework that allows "idle" hardware energy consumption to be eliminated from the software efficiency calculation

Cloud environments usually provide data on CPU / Mem / storage utilisation which could be a starting point for calculating metrics

Application - round 2 - Mo Gandhi

What usage scenarios do we envision and where do we see the highest impact on potential software efficiency improvements?

Can we define software efficiency?

Do we truly understand the trade-offs when calculating efficiency?

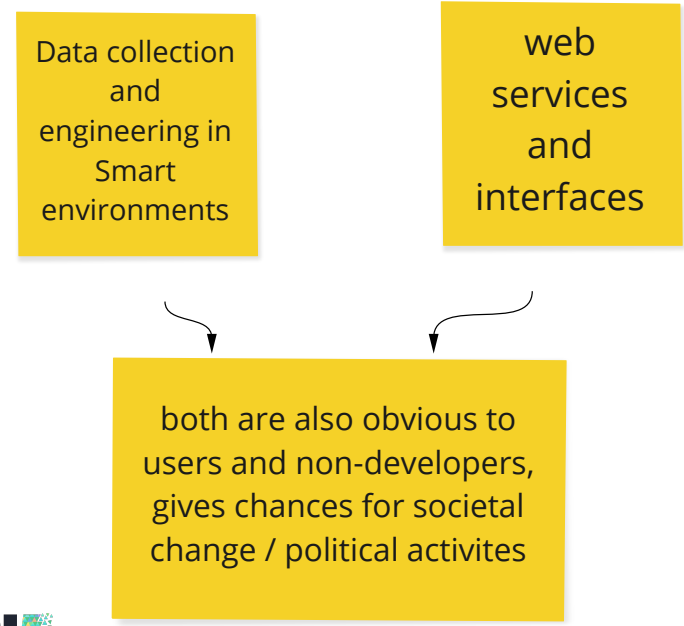
How can we create a metric (or suite of metrics) that incentivise increased efficiency?

How can such a metric be used in conjunction with other metrics within the digital supply chain?

How do we implement such a metric proactively? Is it easy to implement and not just manager speak

Application - round 2 - Erik Albers

What usage scenarios do we envision and where do we see the highest impact on potential software efficiency improvements?



ROOM 3 - OUTLOOK