

# Green Software: Are we on the right track?

---

**Andreas Brunnert**

Professor | Munich University of Applied Sciences HM

Founder | RETIT GmbH

# Session outline



1

## Why?

Why are we measuring anyway?



2

## What?

What do we need to capture, and how?



3

## Where?

Where are our system boundaries?



4

## When?

Should we consider all lifecycle phases?

# Why are we measuring anyway?



## Regulatory Requirements

**ESG & Data Center Reporting:** While reporting traditionally targets facilities directly, allocating footprints specifically to software is vital when separating business units or enabling external providers to expose the footprint of their services.

### Typical Scope

- Data centers (e.g., On-Premise, Cloud, ...)
- External providers (e.g., M365, AI services)



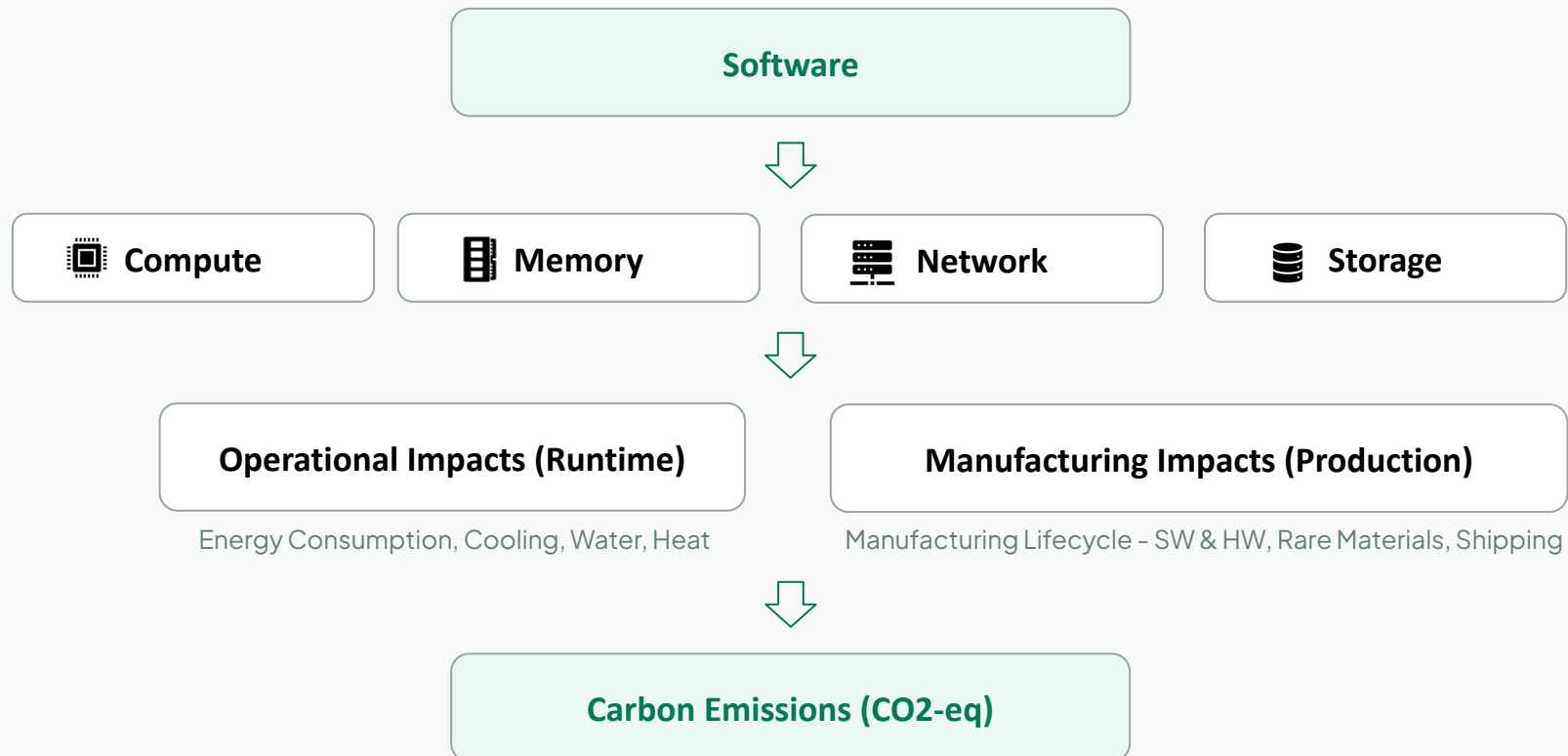
## Optimization & FinOps

**Operational Efficiency:** Merging runtime performance tracking with operational infrastructure budgets to minimize waste directly. Either due to bottom-up initiatives, strategic sustainability goals or market driven.

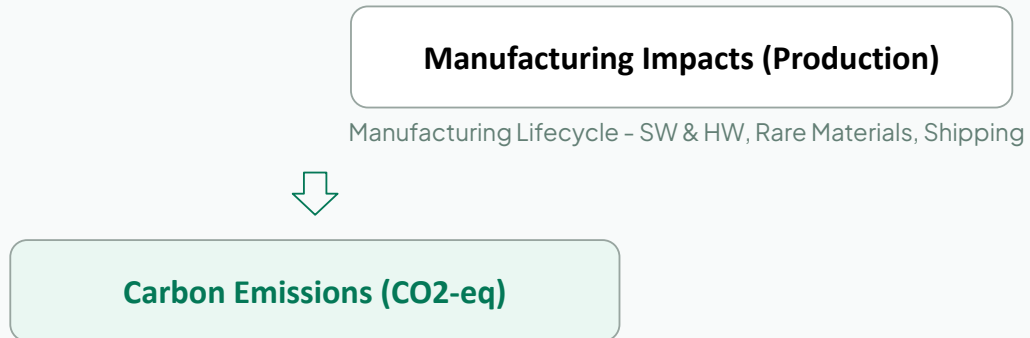
### Typical Scope

- Individual teams
- Individual applications and their components

# What do we need to capture, and how?



# What do we need to capture, and how?



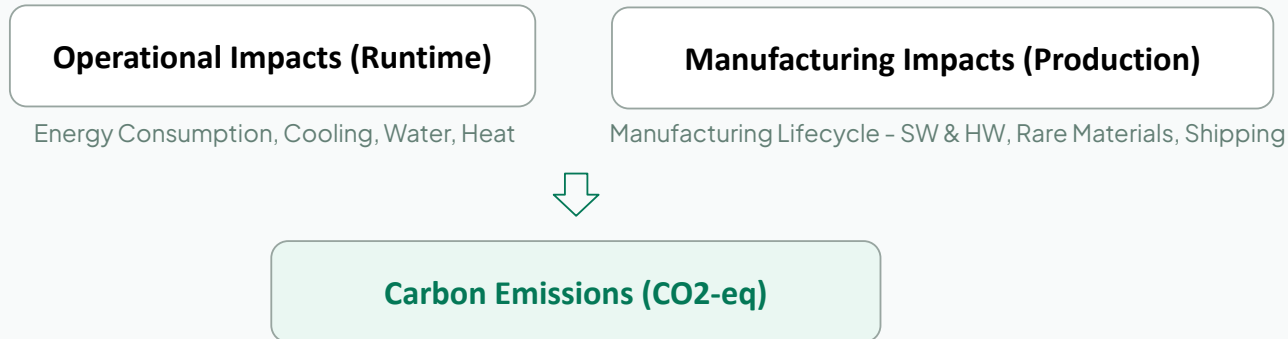
## Hardware Lifecycle Assessment Tools

When a hardware provider doesn't specify their manufacturing impacts, most companies need to resort to LCA tools such as the Boaviztas Datavizta

[dataviz.boavizta.org](https://dataviz.boavizta.org)



# What do we need to capture, and how?



## Cloud Provider Tools

Hyperscale cloud providers deploy, high-level dashboards representing aggregated platform usage metrics:



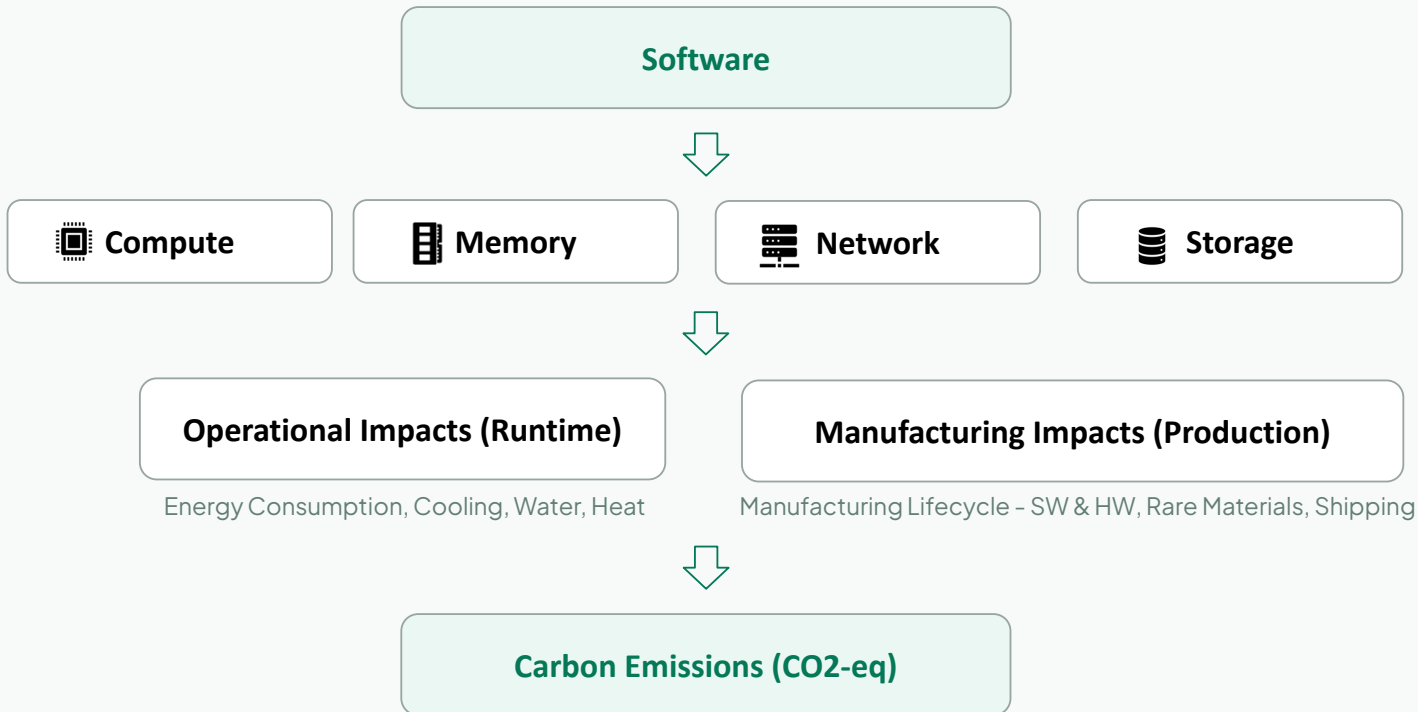
## Hardware Lifecycle Assessment Tools

When a hardware provider doesn't specify their manufacturing impacts, most companies need to resort to LCA tools such as the Boavizta Datavizta

[dataviz.boavizta.org](https://dataviz.boavizta.org)

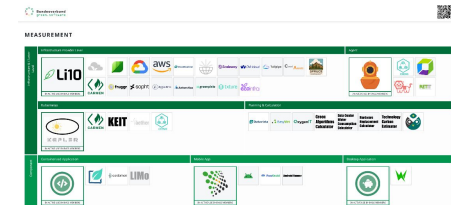


# What do we need to capture, and how?



## Software Energy Attribution Tools

We have a huge amount of tools to attribute energy to software. However, it is hard to find the right tool for your specific purpose.



landscape.bundesverband-green-software.de

## Cloud Provider Tools

Hyperscale cloud providers deploy, high-level dashboards representing aggregated platform usage metrics:



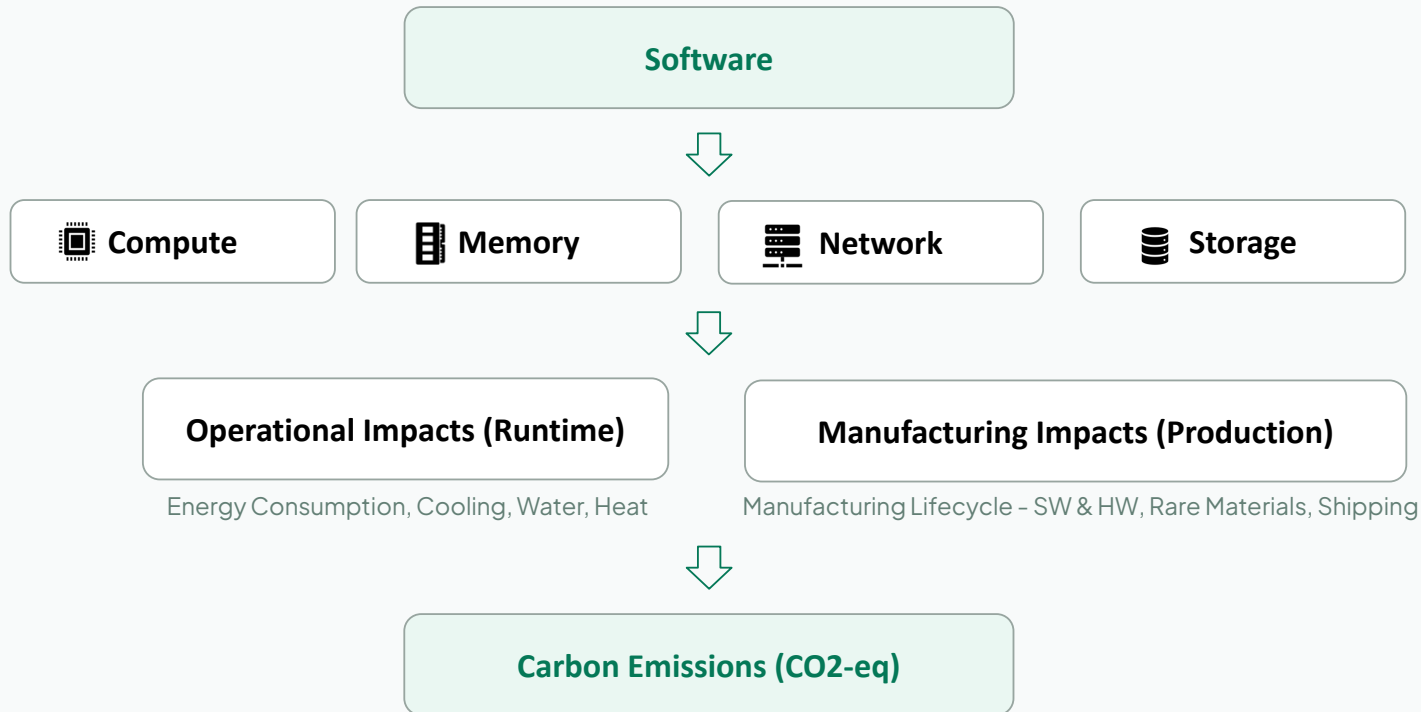
## Hardware Lifecycle Assessment Tools

When a hardware provider doesn't specify their manufacturing impacts, most companies need to resort to LCA tools such as the Boavizta Datavizta

dataviz.boavizta.org



# What do we need to capture, and how?



Directly Measurable by Dev Teams

Beyond Direct Control

## Scope Limitations

-  **Resources:** Teams can precisely measure compute, memory, and network usage but not always at the required level of detail.
-  **Virtualization:** Hard to split shared HW impacts between software components.
-  **Limited Hardware Sensors:** As of today, servers mostly provide PSU measurements (e.g., via Redfish/IPMI) and compute specific measures (via RAPL/NVML).
-  **Indirect Measures:** Carbon metrics (CO2-eq) depend on location and grid intensity and are completely out of control for most IT teams.

# What do we need to capture, and how?

*"When a measure becomes a target, it ceases to be a good measure."*

— GOODHART'S LAW

## Consequences of Indirect Targets

- **Lack of Real Ownership:** Indirect metrics like carbon offsets or regional power updates are easily gamed by shifting server locations (location shifting) without optimizing code.

## What could help?

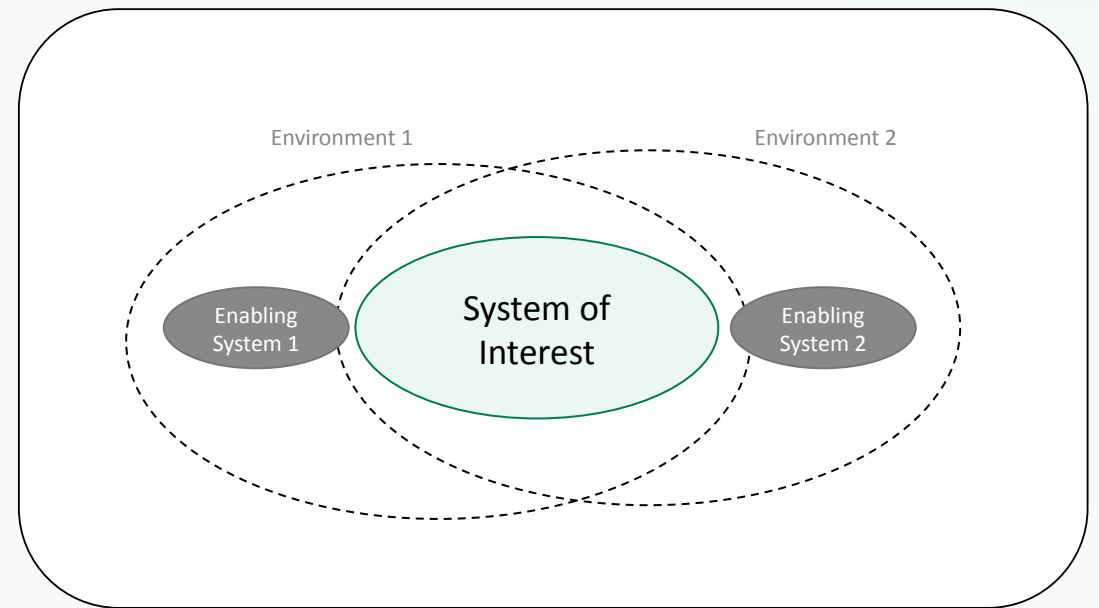
- **Path Forward:** Real, ungameable impact is only achieved by directly reducing actual software resource utilization.

# Where are my system boundaries?

## System of Interest vs. Enabling Systems

When evaluating a software system, we must map our boundaries accurately:

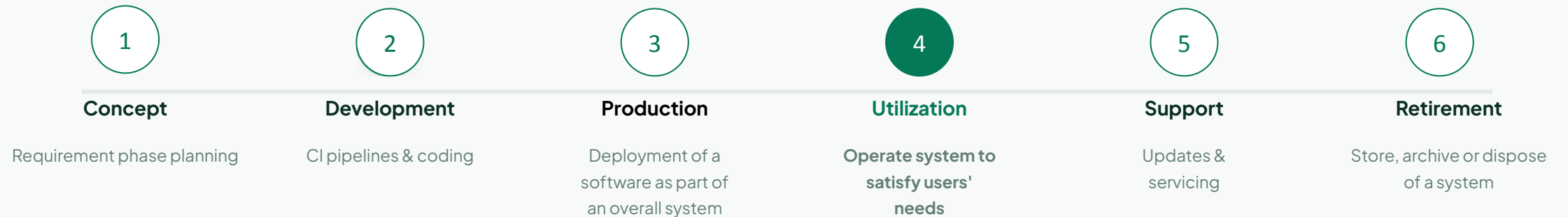
- **Internal vs. Externally Managed:** Who operates which software component?
- **Shared Infrastructure Attribution:** Identifying boundaries of third-party interfaces, database engines, and shared cloud resources that need to be attributed to our system of interest.



ISO/IEC/IEEE 24748-1 - <https://ieeexplore.ieee.org/document/10476385>

# Should we consider all lifecycle phases?

While focus typically falls directly on **Operational Impacts (Utilization)**, software induces footprint throughout its lifetime. How do we divide responsibility across non-operational phases?



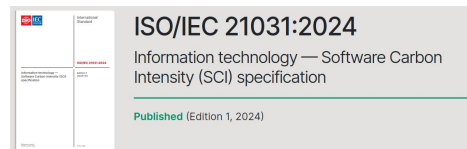
**Software Development Efforts:** If you use standard software such as ERP, how can we split the concept and development effort fairly across all users? Does this need to be updated with each new customer?.

**Missing Standards:** Even when companies try to capture the full lifecycle of software systems, it is hard to compare these numbers. Especially as the boundaries of CI/CD, test systems and support activities are often fuzzy.

# Are we on the right track?

## ✓ Partially, Yes

We possess standard measurements inside highly localized domains such as software energy usage and specifications, such as the Software Carbon Intensity (SCI) for runtime Carbon Accounting or the Blue Angel for Software to include an LCA perspective, but they leave a lot of questions unanswered especially around system and lifecycle boundaries and do not address data collection.



ISO/IEC 21031:2024  
<https://www.iso.org/standard/86612.html>

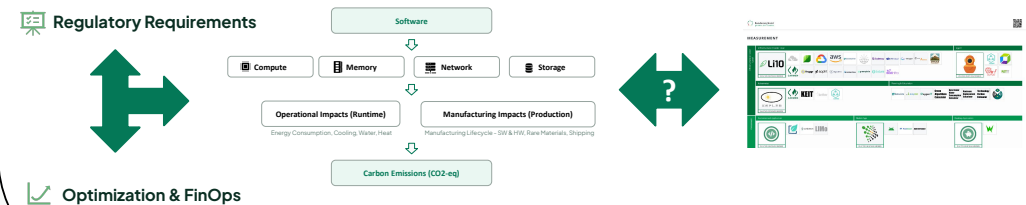
Home // Products // Resources and Energy-Efficient Software Products

### Resources and Energy-Efficient Software Products (DE-UZ 215)

<https://www.blauer-engel.de/en/productworld/software>

## ✗ But Not Seamlessly...

Teams lack easy, end-to-end specifications and support tooling that bridge the intent with the layers between software, hardware and environmental impacts smoothly. Furthermore, the tooling landscape is not aligned with the measurement targets.



# Where do we go from here?



## 1. Clarify the Scope

Establish boundaries for your green software assessment before measuring:

- Define why you are measuring in the first place.
- Define which metrics you can measure and which values need to be calculated.
- Specify the System and Lifecycle boundaries.



## 2. Set Realistic Goals

Avoid developer team fatigue. Protect productivity by focusing exclusively on tangible milestones:

- Focus on resources under your teams control (hardware usage of software and energy use if you can measure it directly) rather than indirect effects such as carbon emissions.

# Where do we go from here?

Closing the gap between intention and impact is a community effort. Connect with lead representatives across these key initiatives here at the conference:



**Amelie**

DIN Specification for Software  
Resource Efficiency  
**What? Where? and When?**



**Anita & Aydin**

Bundesverband Green Software  
Tool Landscape  
**How?**



**Anna**

Blue Angel Certification for  
Software  
**What? Where? and When?**



**Manuel**

Green Software Metrics Research  
**Why? and What?**



# Thank you very much for your attention!

Any questions?



[brunnert@hm.edu](mailto:brunnert@hm.edu)

**Andreas Brunnert**

Professor | Munich University of Applied Sciences HM

Founder | RETIT GmbH