Wapice Ltd.

OPC UA Sparkplug gateway (in Rust language)

Wapice | CREATING A SMARTER FUTURE TODAY

Veli-Pekka.Salo@wapice.com

Wapice, the IoT and AI company from Finland

- Transforming **Industrial and Energy Companies** through digitalization since 1999
- **Private ownership**
- Continuous **organic** and **profitable** growth
- Employing +**350 Software, IIoT** and **AI** experts
- 11 office locations in Finland
- ISO 9001, ISO 14001 and **ISO 27001** certified

National Entrepreneur Award 2016

AAA® Highest Creditworthiness for more than 3 years ©Bisnode 2018

DNV·GL ISO 9001 = ISO 14001

FINNISH SERVICE

THE STRONGEST IN FINLAND PLATINUM 2007-2017

Microsoft Partner — Gold DevOps, Gold Data Analytics, Gold Cloud Platform

ENABLER OF EnergyWeek 2018

FDT Group Member

OPC FOUNDATION MEMBER

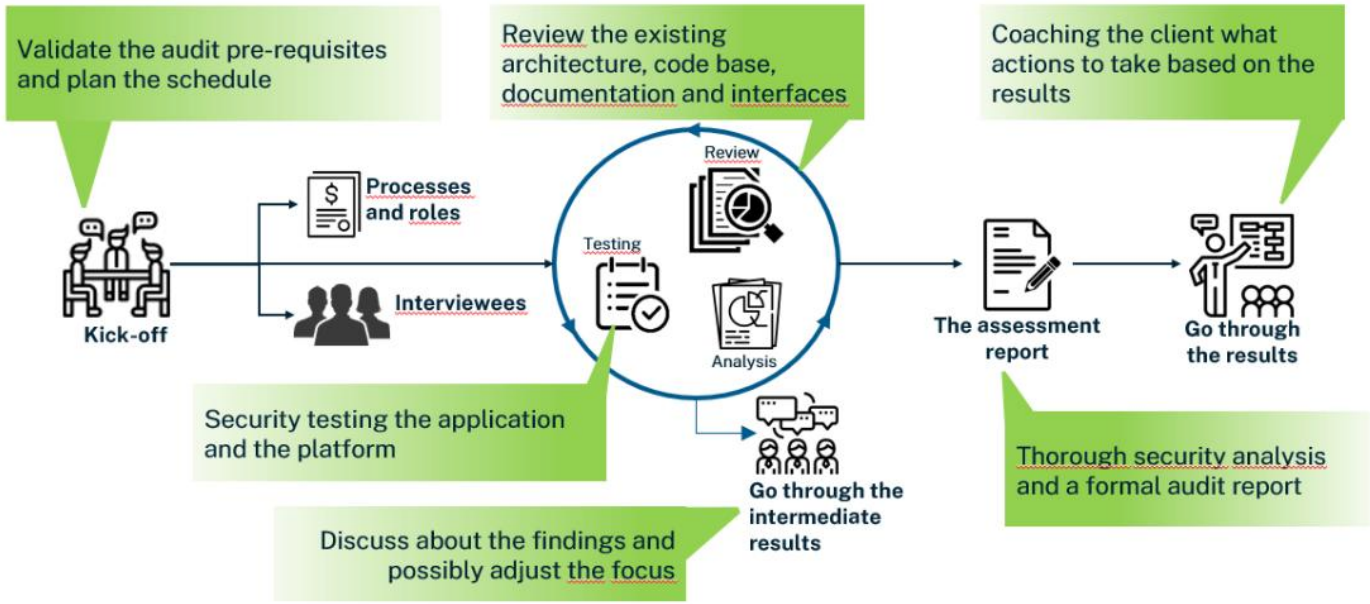
EtherCAT Technology Group

Microsoft Partner — 2019 Partner of the Year Winner, Application Innovation Award

NVIDIA PARTNER

Wapice Security Assesment Service

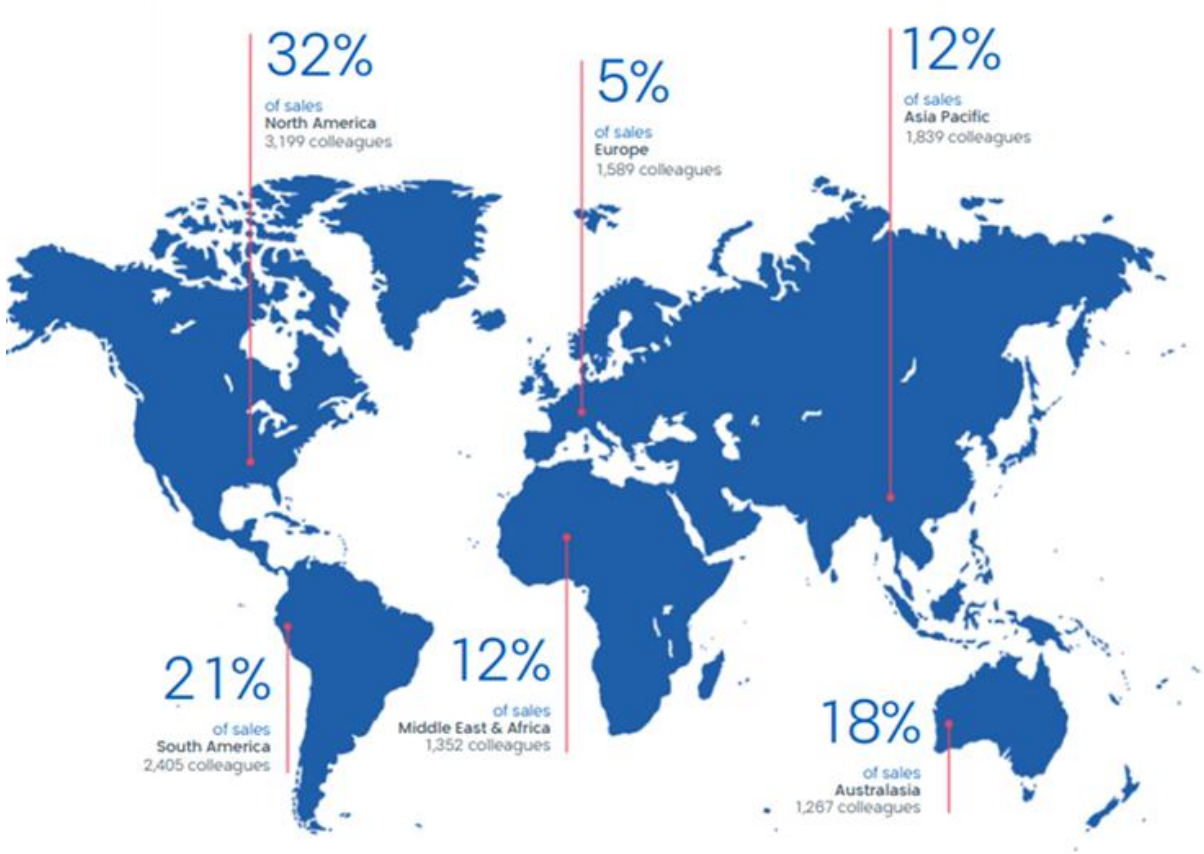
- Reduce risk of loss of customer trust or revenue due to incident downtime and lost sales
- Maintain stakeholder expectations and avoid penalties due to contracts or law
- Reduce incidents that take focus away from the actual business
- Get rid of Fear, Uncertainty and Doubt (FUD).
- Identify any technical gaps relating to regulation like GDPR



- Target: Weir NEXT Intelligent Solutions platform, Edge device, Weir's new global website global.weir

Weir Minerals, Australia

- Global leader in the engineering solutions for the mining and minerals processing industries.
- Products: grinders, pumps, valves, hydrocyclones, screens, and wear-resistant linings.
- Global Presence – Operates in over 70 countries with a robust support network
- Annual turnover approx. 2.9 billion EUR
- 12,000 employees



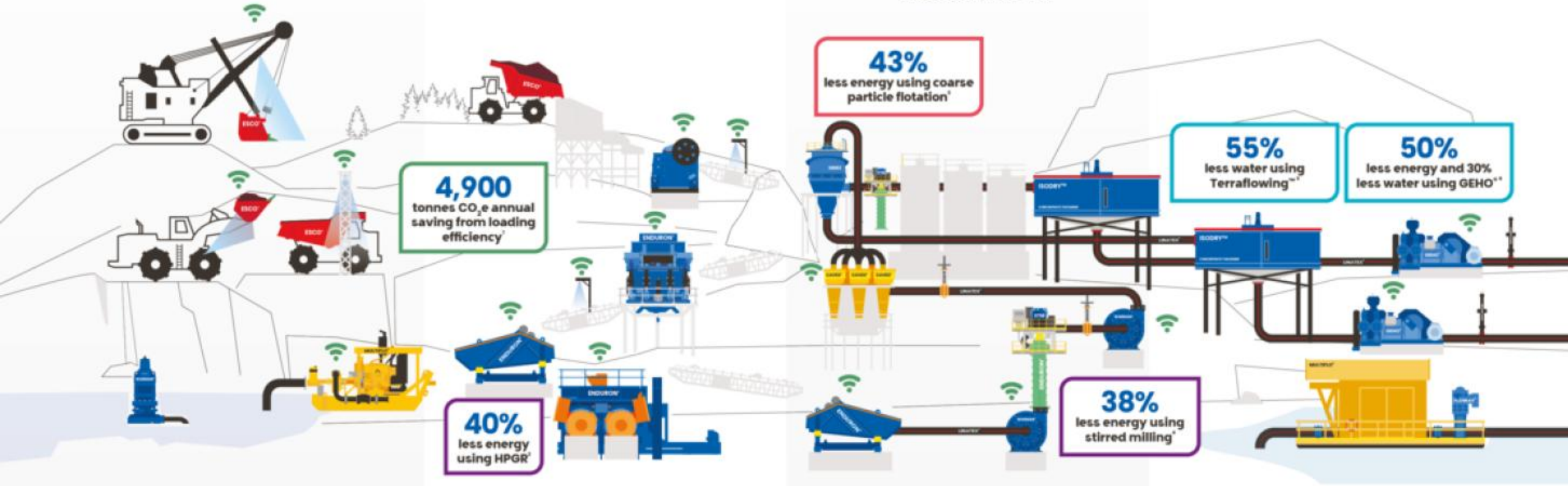
Transformational flowsheets empowered by MOTION METRICS™ and NEXT intelligent solutions

our brand
ESCO

our brand
ENDURON

our brands
WARMAN, CAVEX, ENDURON, LINATEX

our brand
GEHO



Move less rock
Miners want to reduce effort spent on processing zero and low grade ore. We help them optimise the material entering their processing plant.

Use less energy
Mining today is very energy intensive. Our solutions deliver significant energy savings and lower CO_2 emissions.

Use water wisely
Water is fundamental in minerals processing. Our solutions increase water recovery, recycling and introduce water-free steps.

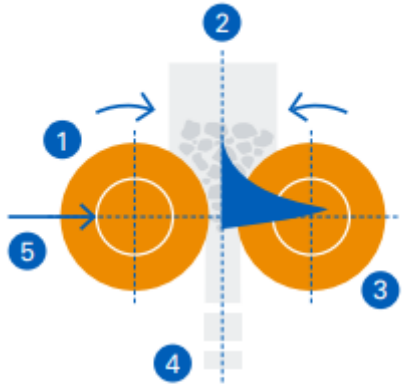
Create less waste
Today, over 90% of waste rock ends up in tailings. We help manage the tailings produced more safely and sustainably.

1. Savings = 3,000 t/y haulage + 1,900 t/y reduced usage of lower priority loaders. 2, 3, 4 & 6. Improvements from Weir's redefined circuit when compared to a conventional circuit. 5. Up to 55% less water compared to a thickener alone.

Source: https://www.global.weir/globalassets/investors/reporting-centre/2025/annual-report/weir-2024-annual-report.pdf

Enduron® High Pressure Grinding Rolls (HPGR)

- Energy-Efficient Crushing
  - Uses high pressure to break ore, reducing energy consumption by up to 40% compared to traditional grinding methods
- Lower water usage
- Long wear life of rolls and bearings
  - Equipped with tungsten carbide studs and a unique roll surface for extended wear life and reduced
- Compact frame design
  - Reduces civil construction costs
- Intelligent Design Features
  - Controlled roller skewing adapts to uneven feed conditions, maintaining pressure and product quality.
- Smart Integration
  - Weir's NEXT Intelligent Solutions (previously Synertrex) for real-time monitoring and predictive maintenance.



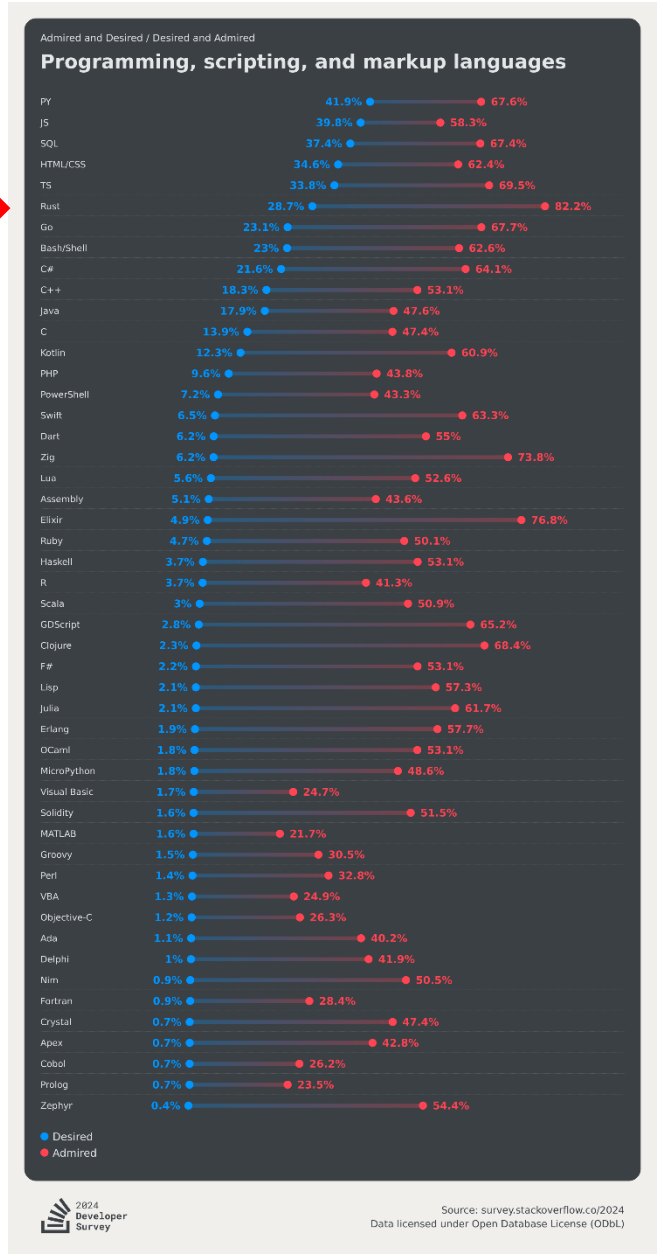
1. Floating roller
2. Feed material
3. Fixed roller
4. Compacted cakes
5. Grinding pressure

Edge Hub Gateway

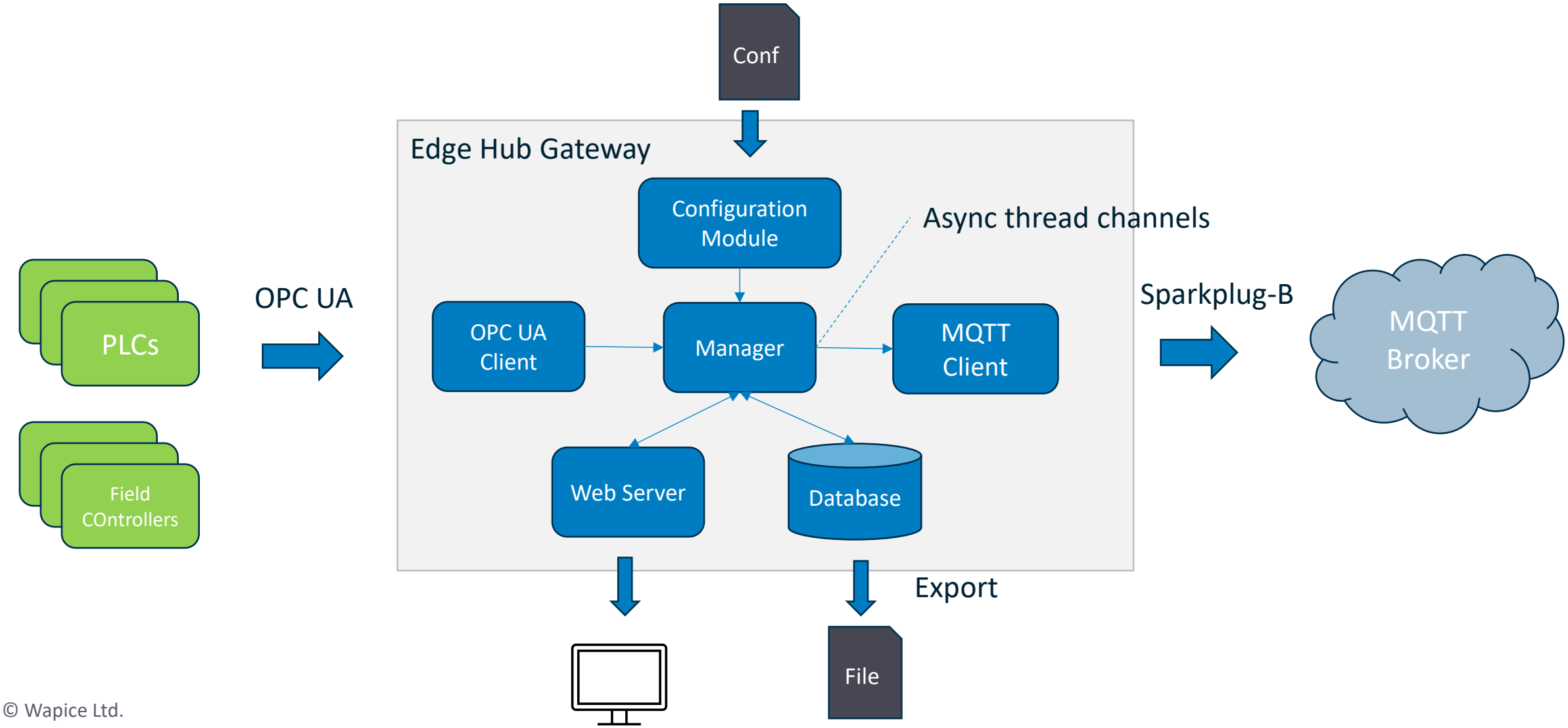
- Purpose of this application is to replace the existing HPGR Diagnostic Box application
- Weir needed a gateway solution for collecting data that is
  - Robust and Cyber-secure
  - High performance
  - Provides simple architecture for collecting data from devices into cloud
  - Supports data format that enhances usage of artifical intelligence
- OPC UA needed for edge-side connection to PLC and field controllers
- MQTT Sparkplug-B was chosen for cloud communication
- Rust chosen as programming language for performance and safety
  - Microsoft: Roughly ~70% of high severity security bugs are the result of memory unsafety

Why Rust?

- Rust was developed as a **fast and memory safe alternative to languages like C and C++**
- **Cross-platform**
- **Fast**
  - Zero-Cost Abstractions: abstractions are designed to be compiled away
  - Compiles down to the same machine code as hand-written low-level code
  - No runtime memory management = no GC pauses
  - Minimal runtime
- **Memory Safety**
  - Rust uses a strict ownership and borrowing system enforced at compile time
  - This prevents common bugs like null pointers, dangling references, and buffer overflows. Garbage collector is not needed.
- **Thread Safety**
  - Rust's type system ensures thread safety by enforcing rules that prevent multiple threads from accessing the same data simultaneously unless it's explicitly synchronized.
- Rich **type system, Debugging** at compile time, Integrated package management, Good IDEs, Documentation, Community, …



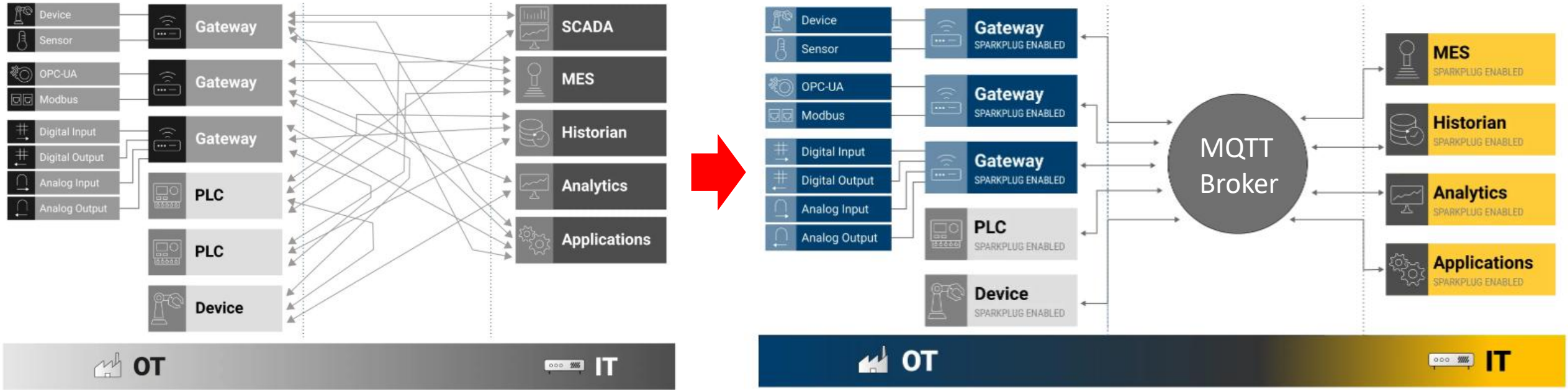
Edge Hub Gateway Architecture



Tokio-rs is built for high performance

- Data updates from 100 ms to 1 second
- Multi-Threaded Work-Stealing Scheduler
  - Tasks are distributed across multiple threads.
  - If one thread finishes its tasks early, it can "steal" tasks from other threads' queues.
  - This leads to efficient CPU utilization and low latency.
- Zero-Cost Abstractions
  - Avoid runtime overhead.
  - Compile down to highly efficient machine code.
  - Be predictable in performance, which is critical for systems programming
- Cooperative Task Yielding
  - Tasks run until they voluntarily yield control—they are not interrupted by the runtime.
  - This avoids preemption overhead and gives developers more control.
  - It also helps avoid race conditions and makes debugging easier.
- Lightweight tasks instead of OS tasks: spawn thousands of async tasks with minimal memory overhead
- Memory-safe, Thread safe, Resistant to misuse

From IIoT protocol spaghetti to Sparkplug B



Source: https://www.hivemq.com/blog/sparkplug-essentials-part-2-architecture/

MQTT Sparkplug-B

- Standardises the MQTT payload and topic namespace
- There was also Sparkplug A specification, but Sparkplug B quickly superseded it, and A is now deprecated
- **spBv1.0/<Group ID>/<Message Type>/<Edge Node ID>/<Device ID>**
- **Version identifier** (currently version 1.0).
- **Group ID**: Identifies a logical group of nodes/devices (e.g., a plant or system).
- **Message Type**: Type of message being sent (e.g., NBIRTH, NDATA, DBIRTH, DCMD, etc.).
- **Edge Node ID**: Uniquely identifies the specific edge node (e.g., a gateway or controller).
- **Device ID**: (Optional) ID of a specific device under the node (used in device-level messages).

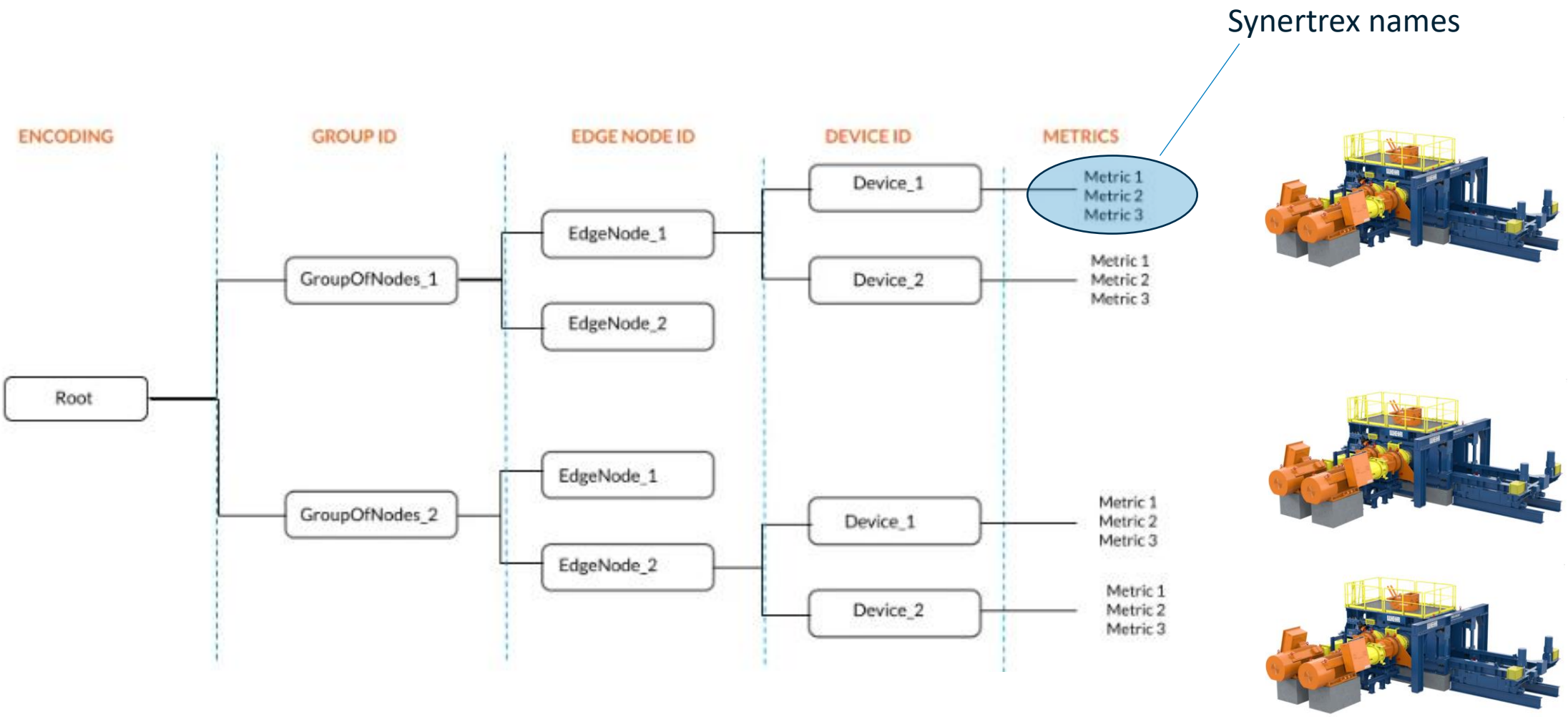
Source: https://www.hivemq.com/blog/sparkplug-essentials-part-2-architecture/

Sparkplug B data payloads

- ✓ **Birth message**: Sent when a device/application comes online. It includes metadata and initial values for metrics.
- ✓ **Death message**: Sent when a device/application goes offline (or is expected to go offline). It typically contains no metrics—just a signal.

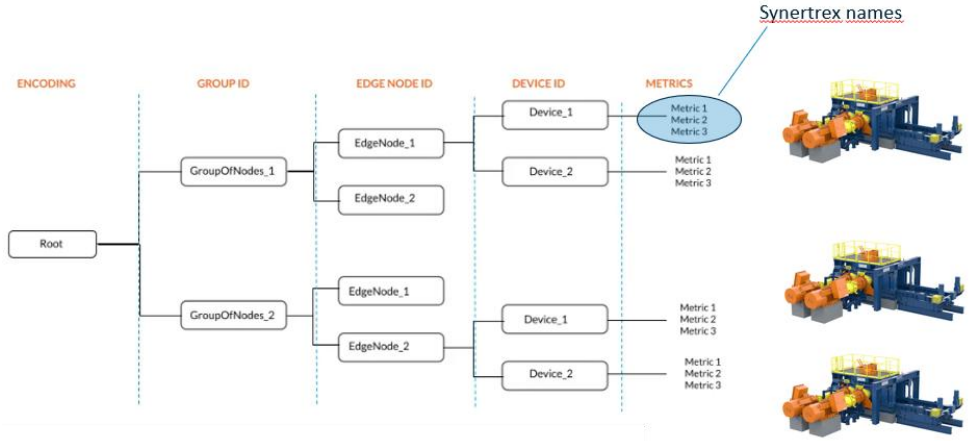
```
{
 "timestamp": 1717843200000,
 "metrics": [
  { "name": "Pressure", "alias": 1, "datatype": "float", "value": 285.0 },
  { "name": "TyreWear", "alias": 2, "datatype": "float", "value": 12.4 },
  { "name": "Skewing", "alias": 3, "datatype": "boolean", "value": true }
 ],
 "seq": 42
}
```

Edge Hub Gateway mapping logic



Unified Namespace (UNS)

- Unified Namespace (UNS) is a “standardized centralized, real-time data structure that organizes and shares industrial information across all systems using a common, hierarchical format”.
- Key features:
  - **Decoupled**: Systems don’t talk to each other directly — they interact through the UNS.
  - **Centralized structure**: All data is organized in a hierarchical, topic-based namespace.
  - **Scalable**: Easily integrates new devices or systems without rearchitecting.
  - **Contextual**: Data is organized by plant, line, machine, and tag — making it meaningful.



Enterprise: Weir
Site: PlantA
Area: Grinding
Line: Line01
Asset: HPGR01
Metric: Pressure
Metric: Temperature
Metric: GrinderWear
Asset: HPGR02
Metric: Pressure
Metric: Temperature
Metric: Grinder Wear

Why is MQTT Sparkplug ideal for implementing UNS?

- **Standardized payload definition and topic namespace**
- **Decoupling of producers and consumers** of data.
- **Report by Exception (RBE)**.
  - Saves bandwidth, memory, and computational power
- **One-to-many communication**
  - Data sent once, received by multiple receivers.
- **Flexibility**
  - Add/remove devices and applications without affecting the system.
- **Security**
  - TLS encryption, authentication, and structured Protobuf payloads.
- Data governance: **Centralized permission and policy handling**.
- Discovery: **Devices and applications auto-discover data and topics** through birth and death messages

Key takeaways

- OPC UA vs Sparkplug as cloud communication protocol
  - OPC UA's role in field level communication is rock solid
  - Standards are needed for cloud communication as well, but they must be lightweight – Data volume directly impacts cloud costs—especially in large-scale industrial environments with high-frequency telemetry
  - In our opinion Sparkplug B over MQTT is ideal cloud communication protocol
  - Reduces complexity, accelerates deployment, and enhances AI-readiness
- Rust
  - Learning curve in Rust is steep, but AI assisted coding helps in learning and makes Rust "more available". ☺
  - We expect more and more projects to adopt Rust in future

Wapice

CREATING A SMARTER FUTURE TODAY

Visit **wapice.com**