



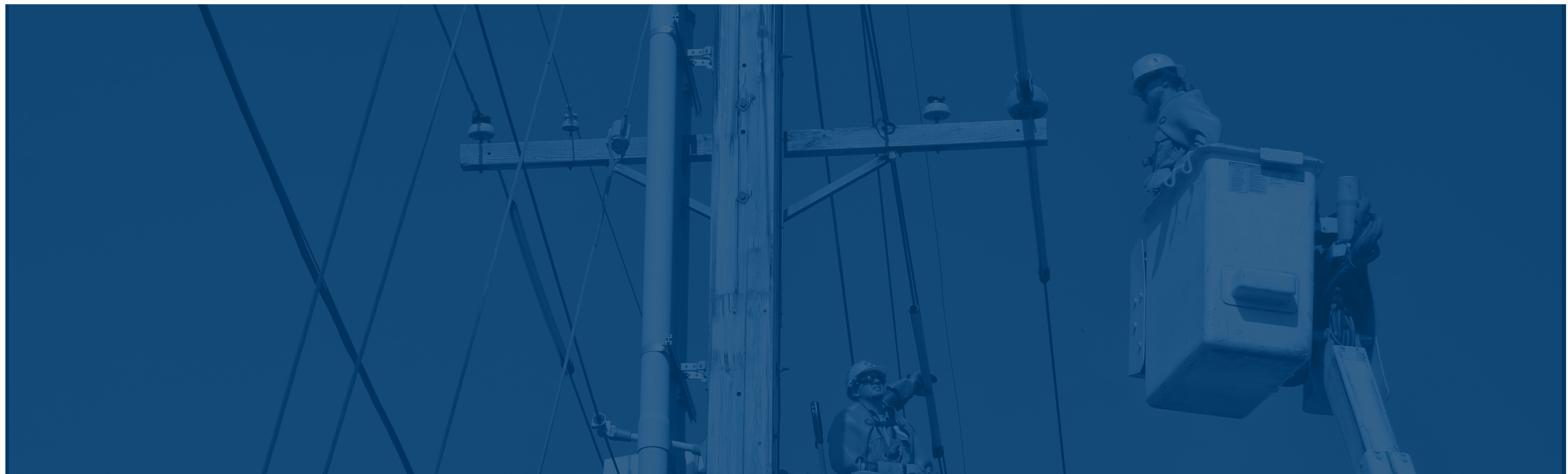
OLF ENERGY

SEAPATH

Virtualization for Real-time Power Grid Substation Automation

About Seapath

SEAPATH is the result of a collaboration between 2 major european electricity players and Open source expertise.



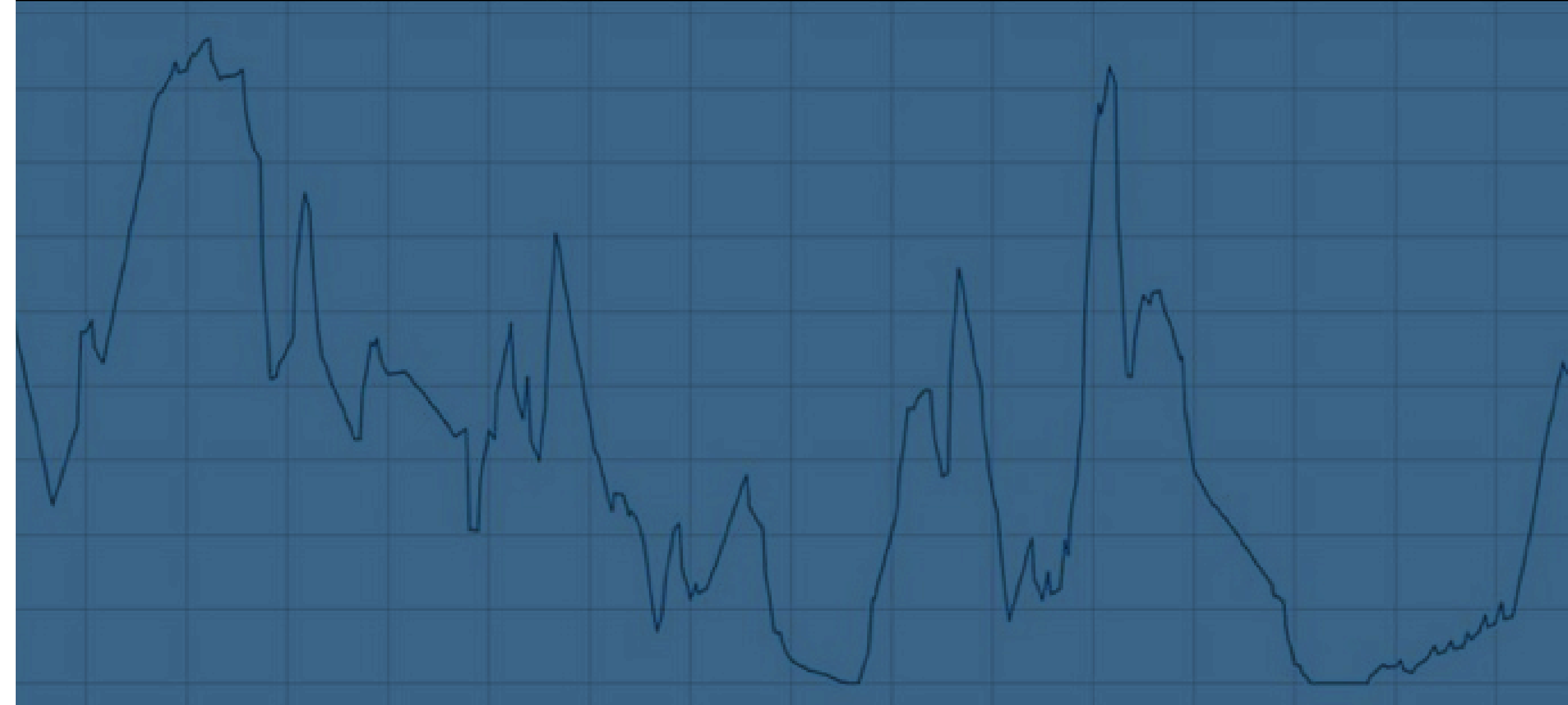
Context

Energy Transition drives change in power transmission and distribution grids

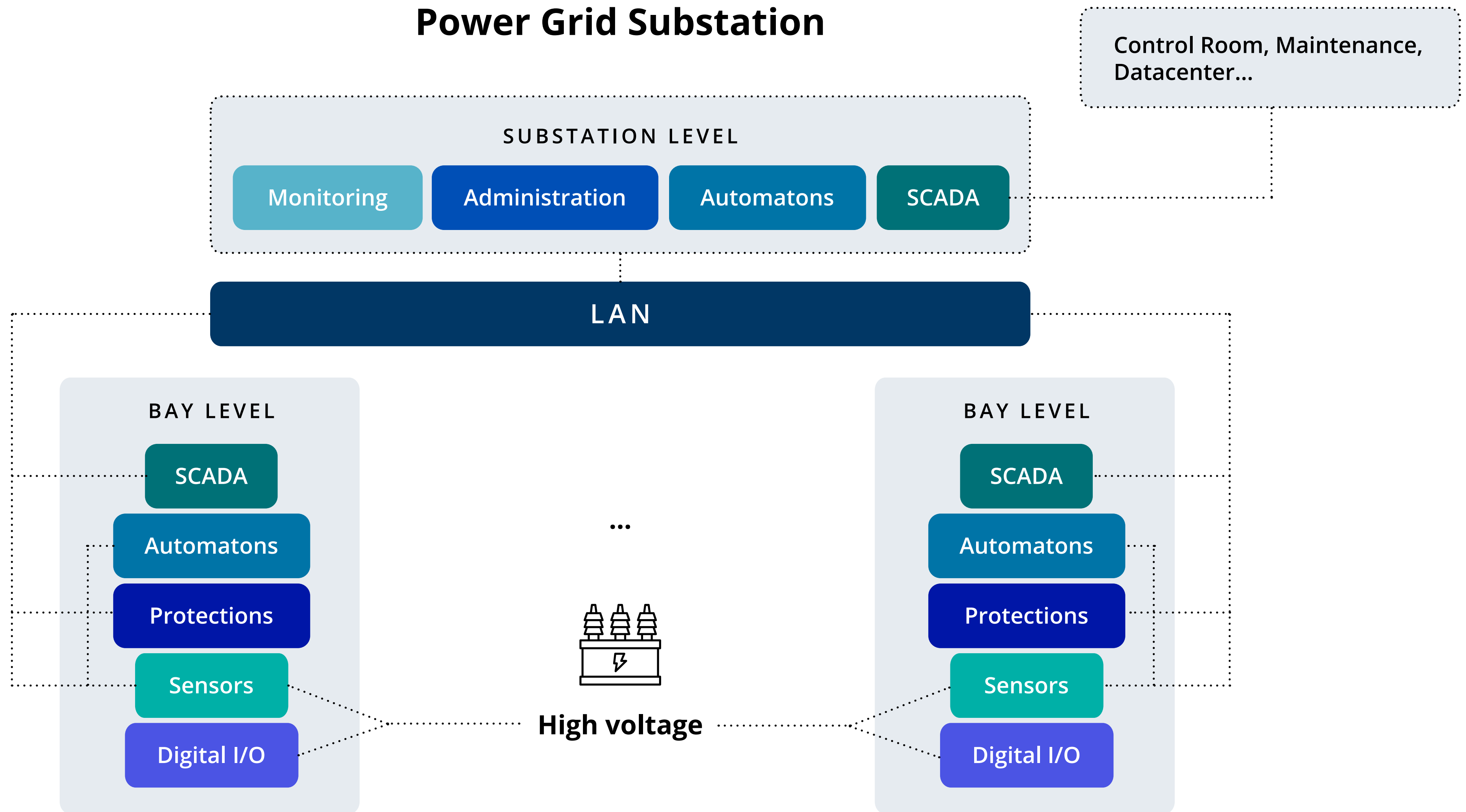
- Distributed renewable energy sources
- Demand response
- Electric mobility
- Smart services to the grids from a growing number of third-parties

Need to **swiftly adapt grid control architectures**

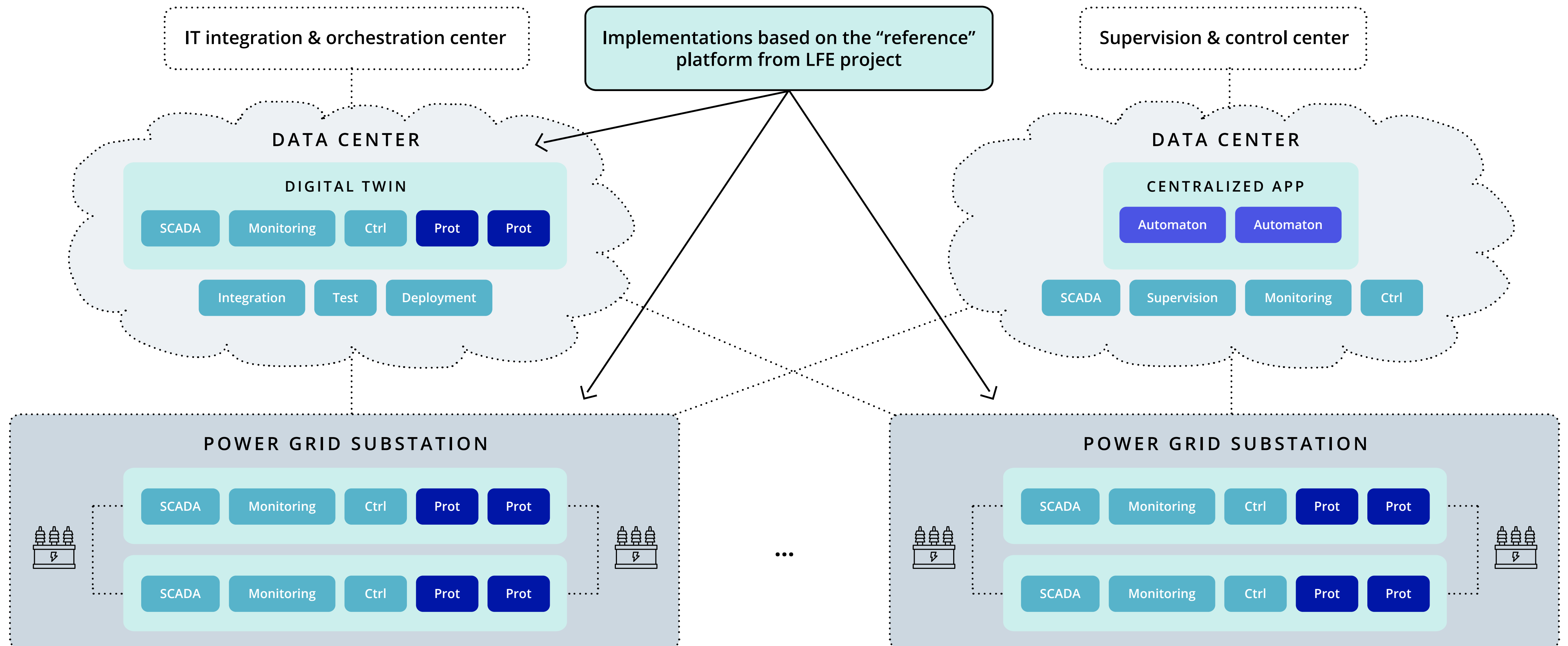
- Multiplication of distributed controls
- More dynamic and adaptive automation functions
- Increased data management needs



From where we start (digital substation)



Virtualization for new grid control architecture



The background of the slide is a photograph of a high-voltage electrical substation, featuring several large insulators and metal support structures. The entire image is covered with a semi-transparent blue filter. The text 'SEAPATH' is centered in the upper half of the image in a white, bold, sans-serif font.

SEAPATH

Software Enabled Automation Platform
and Artifacts THerein

The goal of Seapath



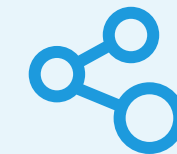
Develop

a **reference design** and
industrial grade open source
real-time platform



Host

virtualized **automation**
and protection
applications



Share

a platform between **multi-**
provider applications
(hardware agnostic)



Combine

performance and
safety

The background image shows a complex industrial facility, possibly a refinery or chemical plant, at night. The scene is dominated by tall distillation columns, intricate piping, and structural steel frameworks. The entire image is overlaid with a semi-transparent dark blue filter. Numerous bright, multi-pointed starburst light effects are scattered across the scene, particularly concentrated around the upper sections of the distillation columns and various points along the piping, giving the impression of sparks or intense heat. The overall mood is industrial and high-tech.

Vision & technical concept

Technical requirements



Virtualization



High availability



Software Defined Network



Cybersecurity



Minimal services / configurations



Low network latency

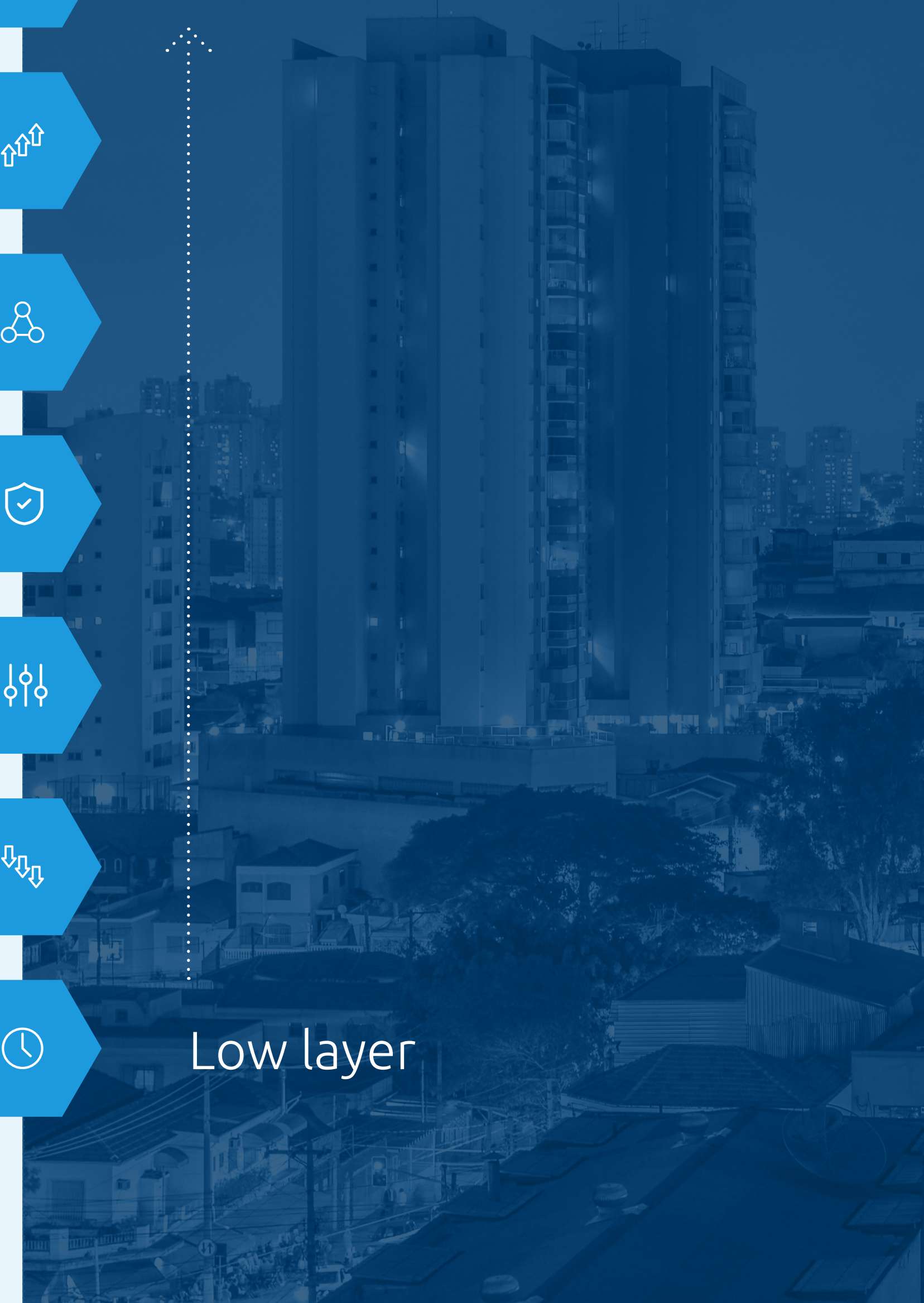


Real-time



High layer

Low layer



Existing solutions

Off-the-shelf solutions providing either

- HA & Virtualization platform
- Real-time platform
- Low-latency platform

But not a mix of that !

- ⊕ Minimal firmware
- ⊕ Highly configurable



Let's create our own Linux distribution without reinventing the wheel !

The Yocto Project



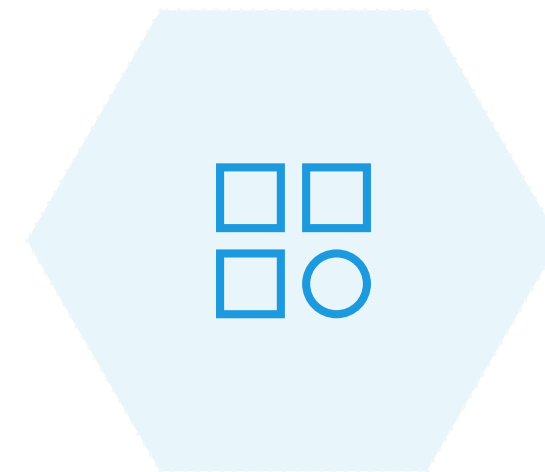
Collaborative Linux Foundation project.
Allows to:



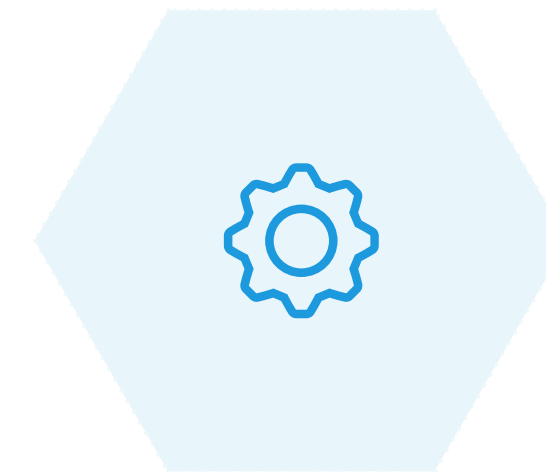
Create custom Linux
based systems from
source code



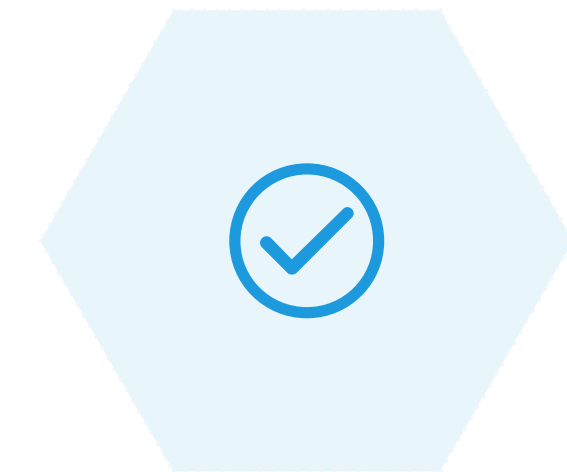
Be agnostic of the
hardware architecture



Aggregate tons
of Open Source
components (including
Virtualization and HA)

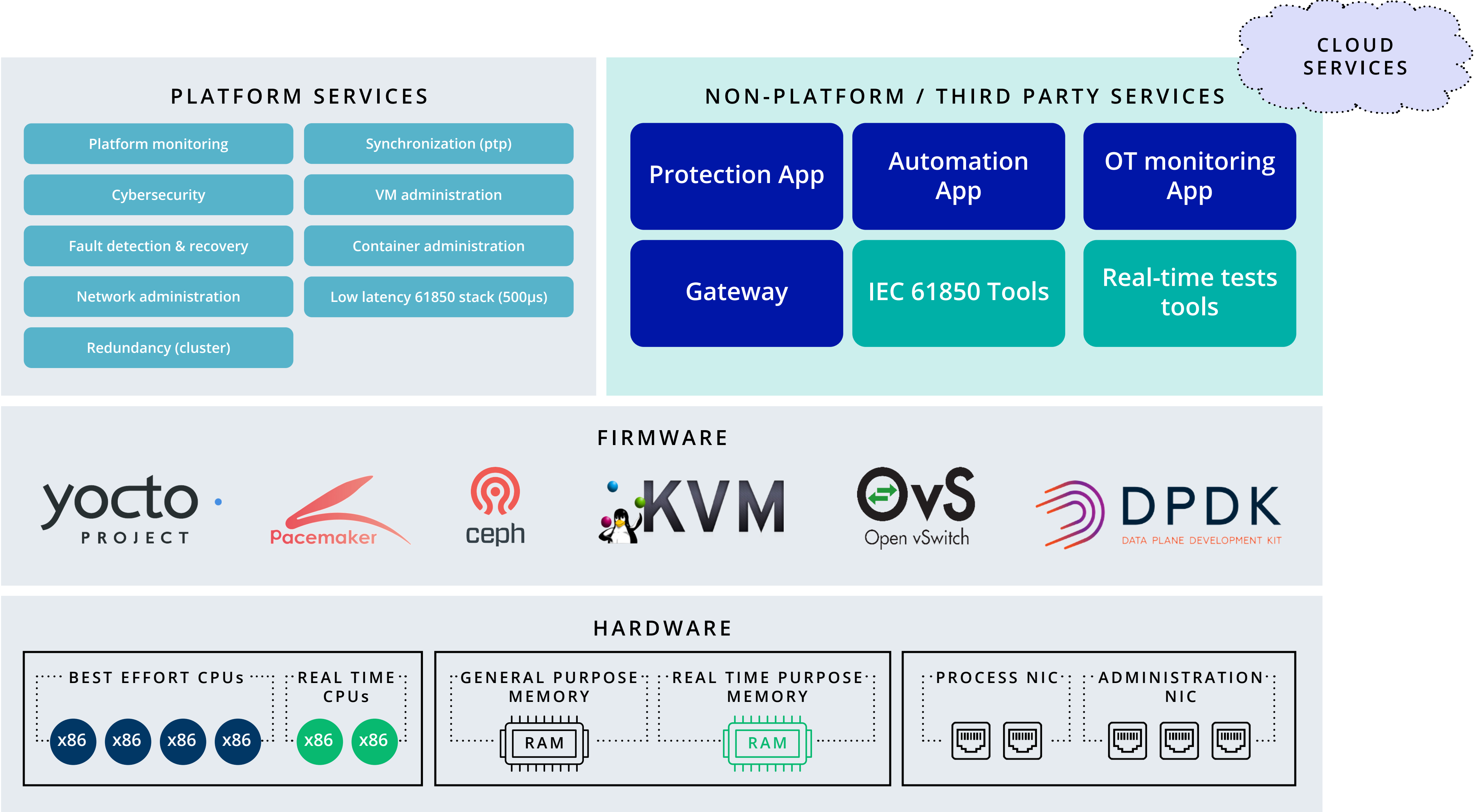


Configure, modify
each component



Check / patch
Common
Vulnerabilities and
Exposures (CVE)

Technical architecture and stack



The background of the slide is a deep blue color with a subtle, high-resolution texture that resembles the surface of the ocean or a satellite view of a coastline. The texture consists of fine, swirling patterns and variations in shade, giving it a sense of depth and movement. In the center of the slide, the words "Ensuring reliability" are written in a clean, white, sans-serif font. The text is centered both horizontally and vertically, standing out clearly against the dark blue background.

Ensuring reliability

Continuous Integration

- **To speed-up adopting new technologies** in a critical infrastructure environment that is growing in complexity, **a shift-left is needed.**
- **Using containerization and virtualization**, combined with infrastructure-as-code enabling technologies (such as ansible) **will allow automated DTAP environments.**

- **Early testing** (fail early) in such an environment will help **to create a streamlined development process.**
- Such technology will **limit start-up cost for new innovations, and open up the industry to newcomers.**

SEAPATH CI building blocks

1

SEAPATH recipes (including
docker and Virtual Machine)
in github



2

**Automated image build
process** including unit tests
provided by the Yocto project



3

**Automatic build
pipeline** and orgistration
by Jenkins



4

**Automated deployment
and configuration**
by Ansible



5

Platform integrity tests
using Cukinia



6

Integration testing
of typical substation
applications



7

Real-time test bench
for evaluation of real-time
requirements

... (interoperability testing, Site Acceptance Test, ...)



Proof of Concept

Latency checking (cyclic test): Host

Configuration:

Operating system:

Image based on yocto including KVM, Pacemaker
Corosync and Ceph, Kubernetes, OvS-DPDK, Docker.

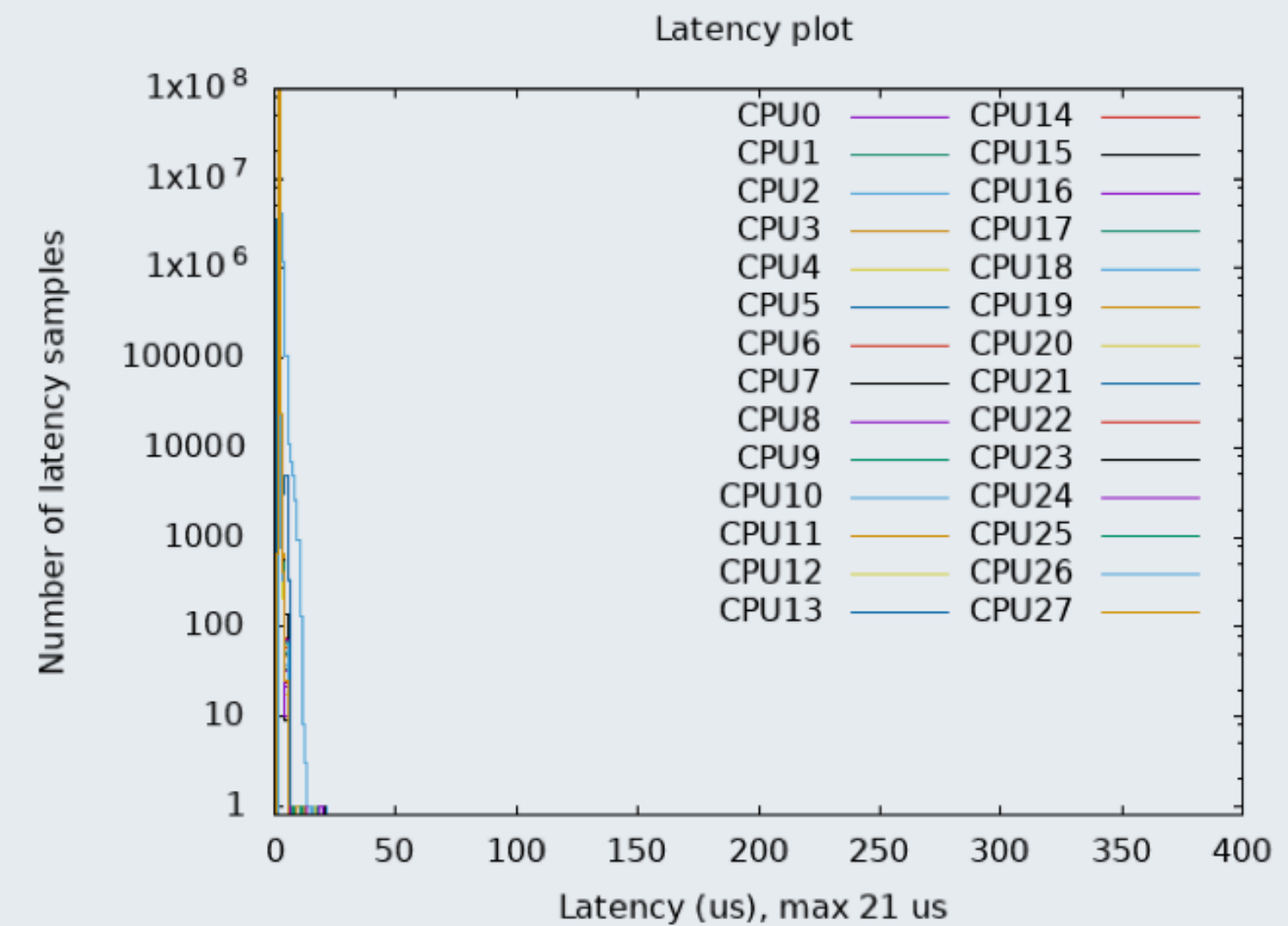
Kernel: 4.19 (preempt-rt and non preempt-rt)

Hardware:

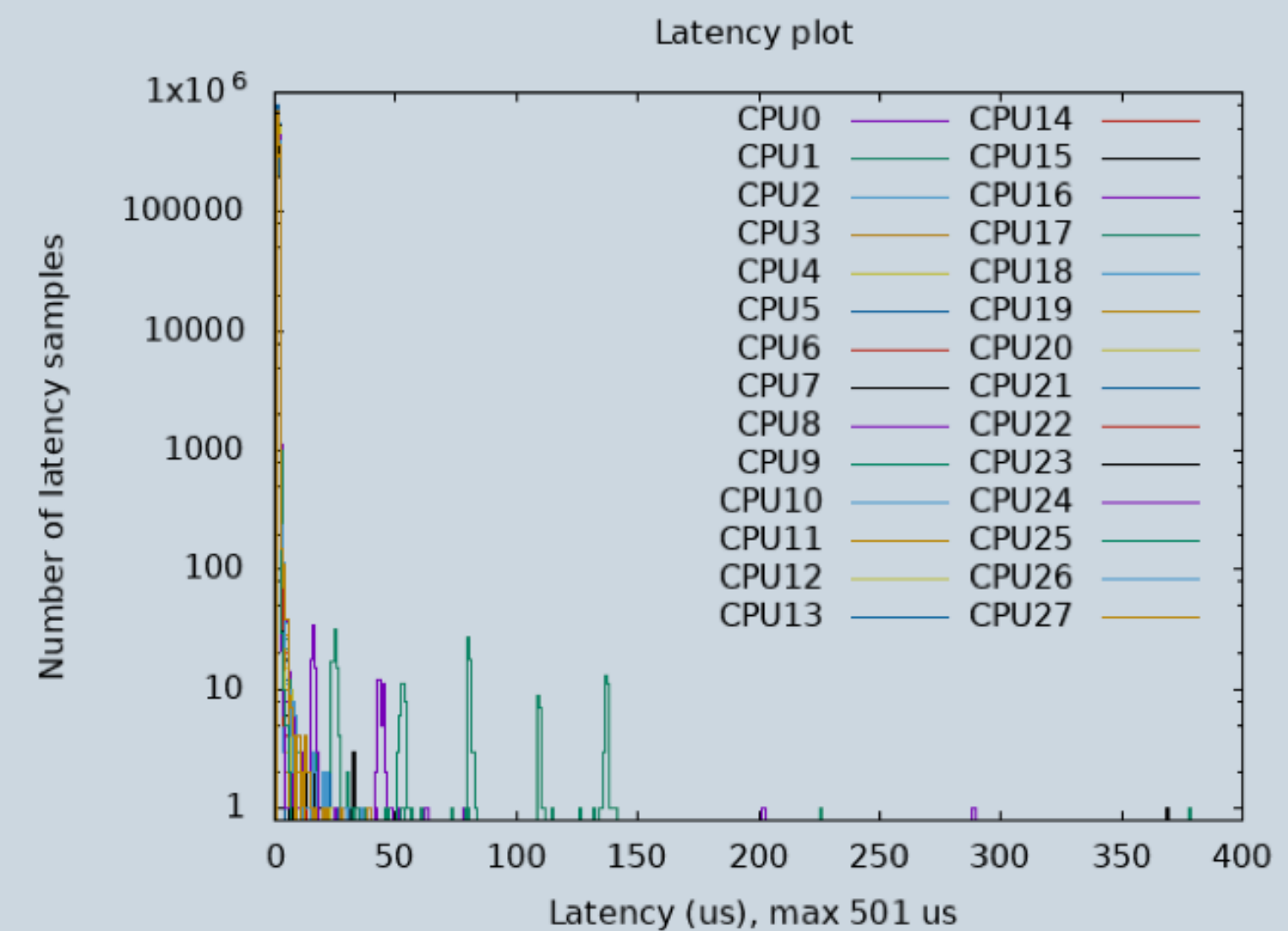
CPU: 14 Core E5-2680V4

Memory: 32GB DDR4

Circuit Board: ASMB-8231-00A1E



Preempt-rt



non Preempt-rt

Latency checking (cyclic test): Guest

Configuration:

Host Operating system:

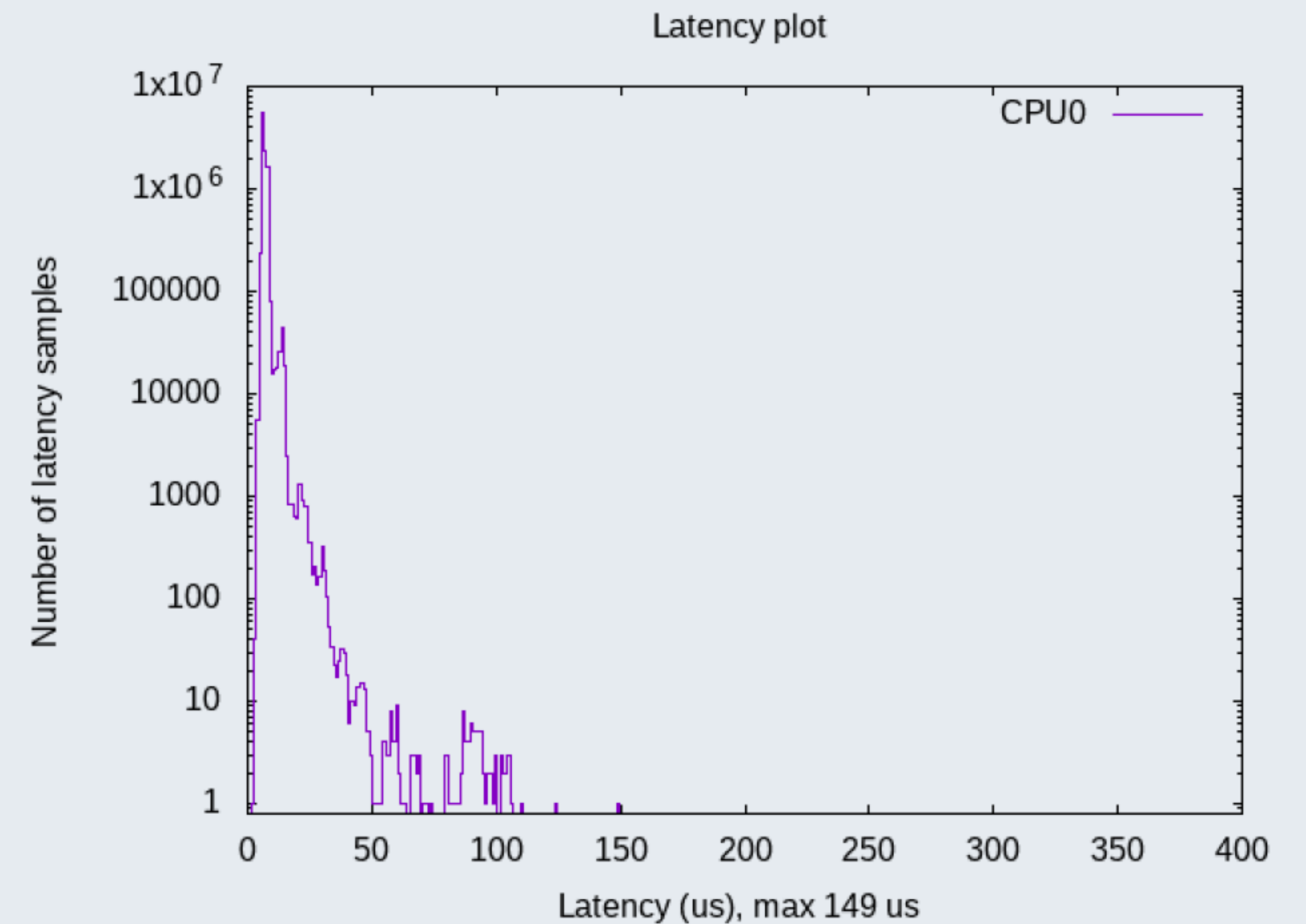
Host: see previous slide

Kernel: 4.19 preempt-rt

Guest:

Same as host without kvm, Ovs-DPDK

Kernel: 4.19 preempt-rt

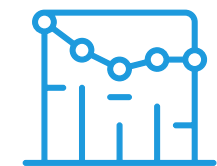


Real-time testing SEAPATH

It will allow anyone to:



Ensure a specific set-up has the real-time characteristics suitable for its critical application in the substation.



Benchmark different hardware platforms to see if it can meet all the required performance criteria.

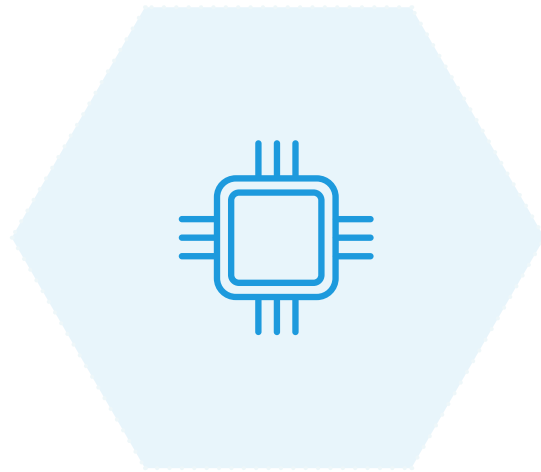


Understand the scale-ability potential of different setups.



Validate the real-time performance-impact of any configuration, or **changes** made during development and operations.

Real-time testbench



Required hardware

Measurement hardware:

PC with 2 NIC's, 1 DPDK supported network card



Required software

MoonGen

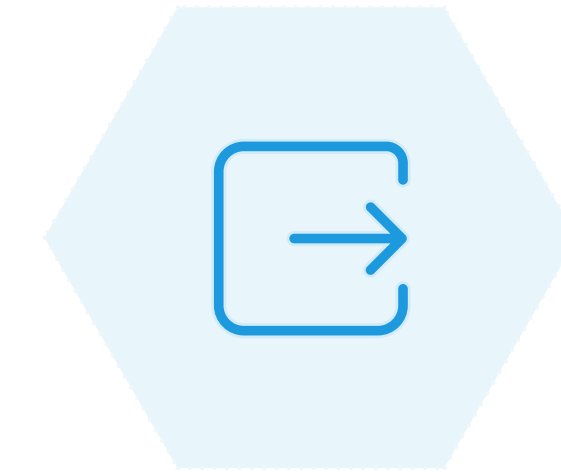
<https://github.com/emmericp/MoonGen>

TestBench

<https://github.com/robidev/moongen-rfc2544>

Test applications

<https://github.com/seapath/meta-seapath-tools>



Output

Report of performance measurements,
"inspired" by RFC2544

Example test report

RFC 2544 Test Report

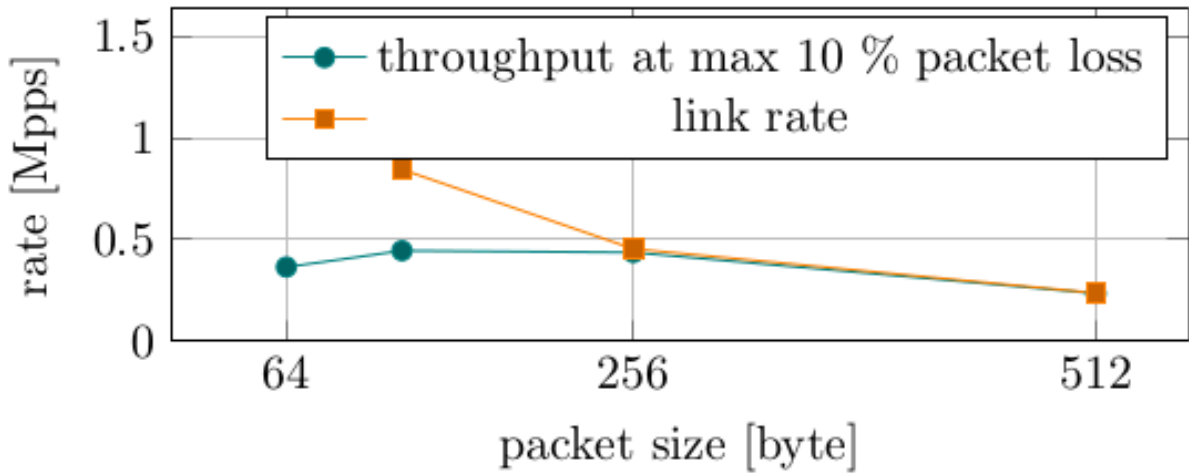
1 General Test Information

Device Under Test: KERNEL
Operating System: votp
Date: 2021-03-22

2 Throughput

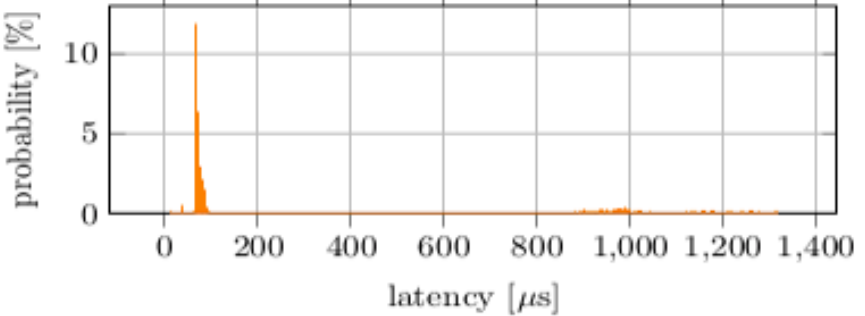
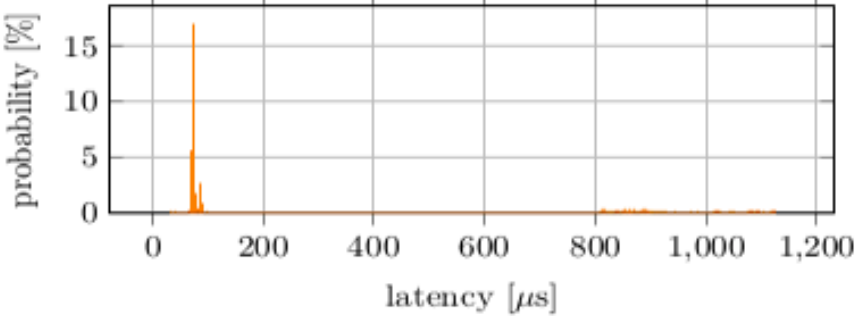
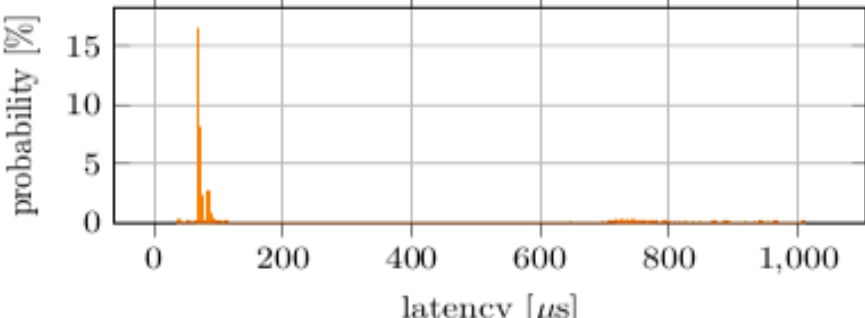
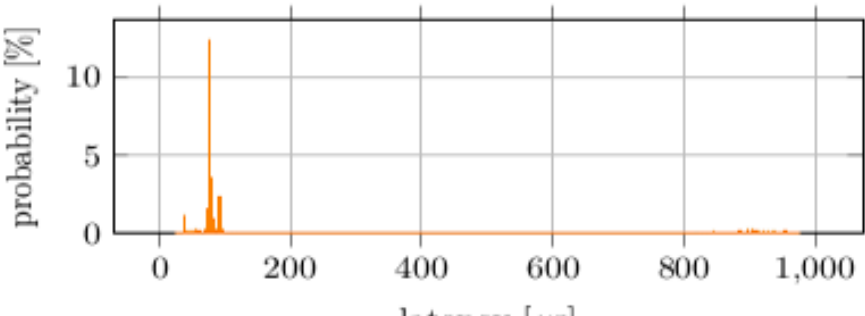
Test Duration: 1 s
Maximal Loss Rate: 10.000 %
Accuracy: 100 Mbps

Frame Size (bytes)	Iteration	Total Tx Frames	Total Rx Frames	Throughput (Mpps)	Throughput (Mbps)
64	1	363384	363384	0.363	244.194
128	1	444150	444150	0.444	525.874
256	1	435393	435393	0.435	961.348
512	1	233226	231432	0.233	992.610



3 Latency

Test Duration: 1 s

Frame Size (bytes)	Throughput (Mpps)	Latency Min (μ s)	Latency Avg (μ s)	Latency Max (μ s)	
64	0.360	11712.0	193497.4	1314224.0	
128	0.440	31984.0	199184.2	1123120.0	
256	0.433	35024.0	187931.2	1009152.0	
512	0.233	25160.0	194260.6	978208.0	

Call to Arms

Desired contributions:

- We need end user feedback!
- End user applications
- Feedback regarding the platform's features (do we miss any?)
- Test bench validation, and checks for completeness



<https://github.com/seapath/>

Thank you for your attention

<https://github.com/seapath/>

OLFENERGY