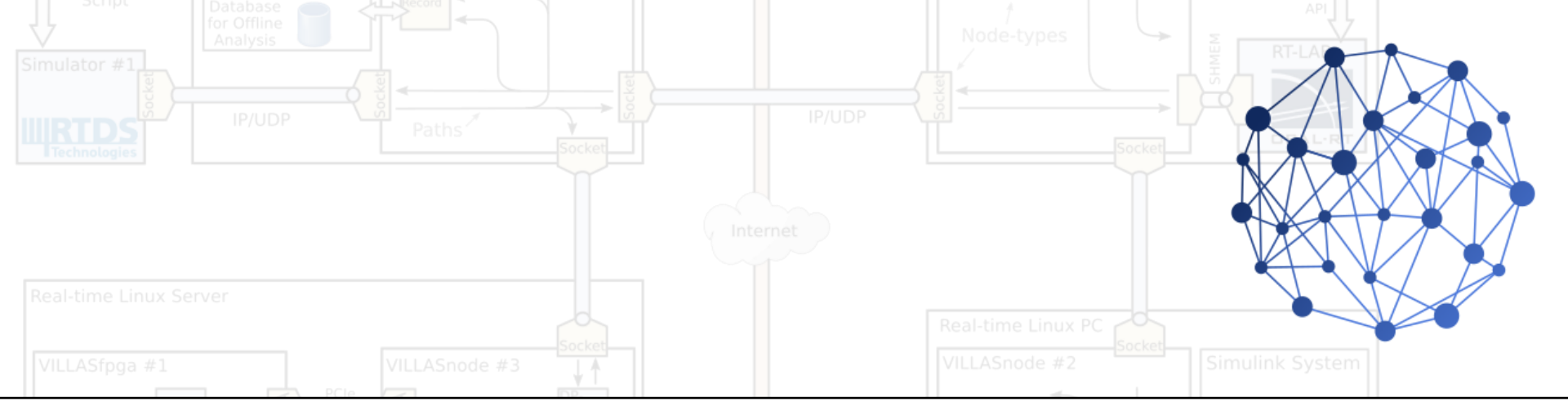


VILLASframework

A framework for
Virtually **I**nterconnected **L**aboratories for **LA**rge systems **S**imulation/emulation

Institute for Automation of Complex Power Systems (ACS)
RWTH Aachen University
Germany

Steffen Vogel, M.Sc.
(v4 – 18.05.2017)

ACS | Automation of Complex Power Systems
E.ON Energy Research Center
RWTH AACHEN UNIVERSITY

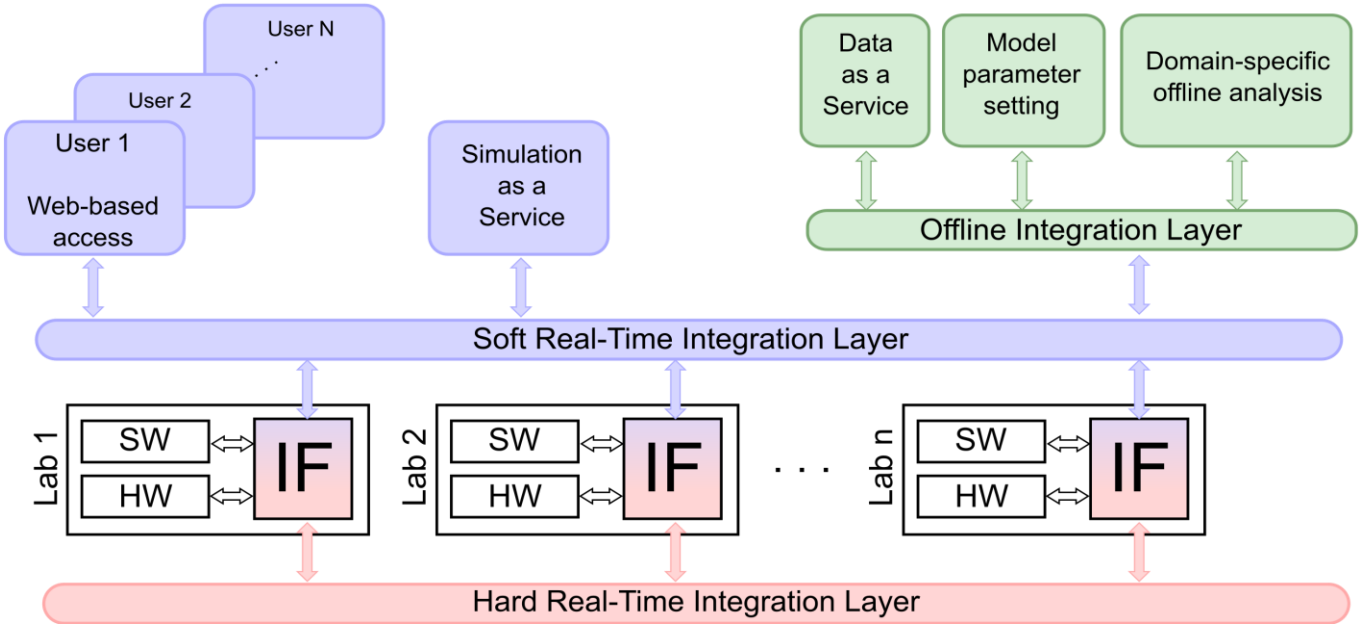
VILLAS
Framework for virtual integration of laboratories for co-simulation

- VILLAS project - Virtually Interconnected Laboratories for LArge systems Simulation/emulation
  - a flexible integration of the resources available at each laboratory
  - a flexible utilization of an infrastructure as a whole

- VILLAS framework aims at providing a specific set of interfaces and services
  - hard and soft real-time interfaces
    - = integration of geographically dispersed hardware and software assets for joint operation in a single experiment
  - high-level interfaces such as a user interface, an interface for data logging
    - = interactions with an experiment and post-processing of results for further analyses
  - high-level services, such as Simulation as a Service, Data as a Service
    - = flexible access for third parties to leverage utilization of the infrastructure

VILLAS
Integration Layers

- Hard Real-Time Integration Layer (H-RTIL)
- Soft Real-Time Integration Layer (S-RTIL)
- Offline Integration Layer (O-IL)

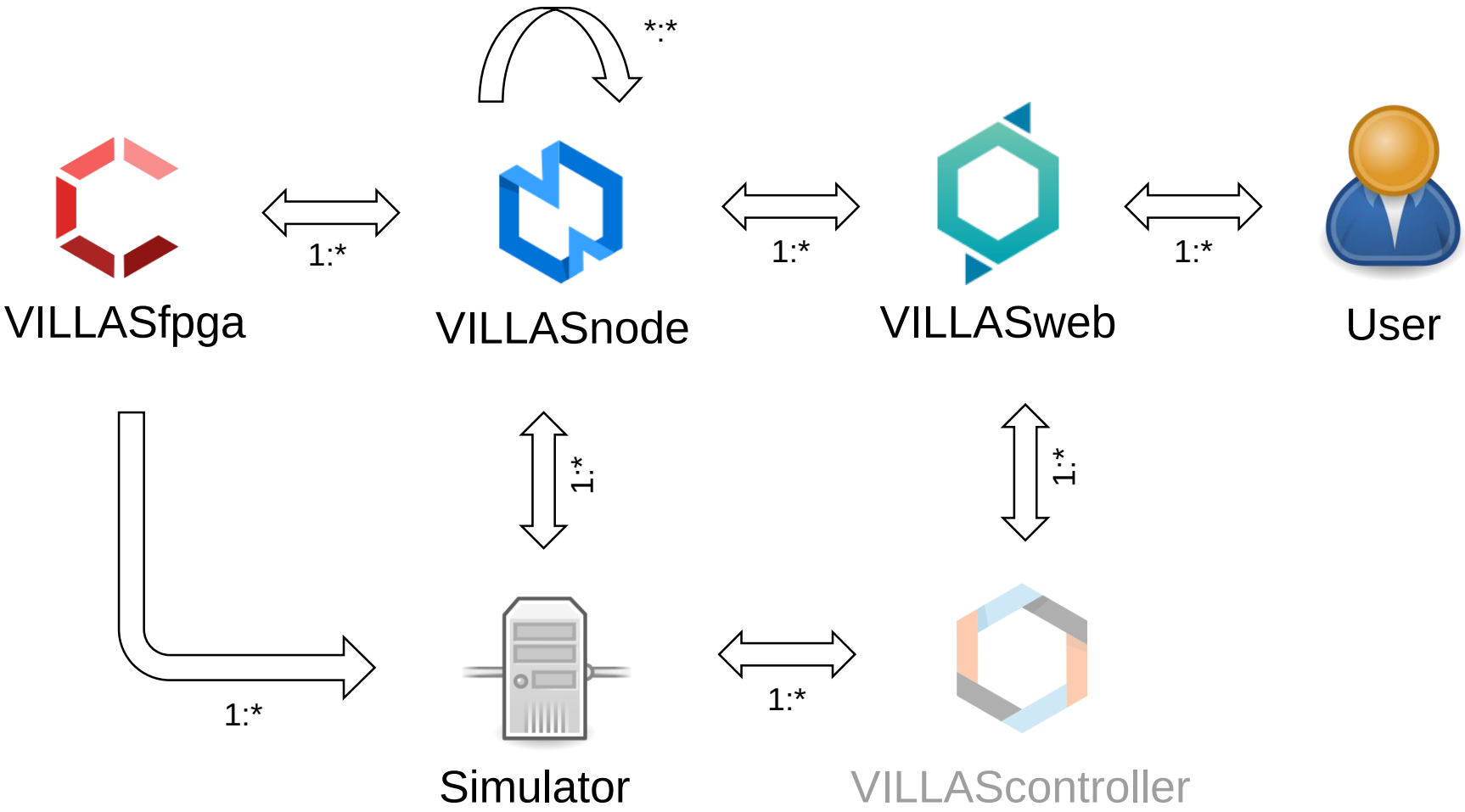


VILLASframework
High-level architecture overview

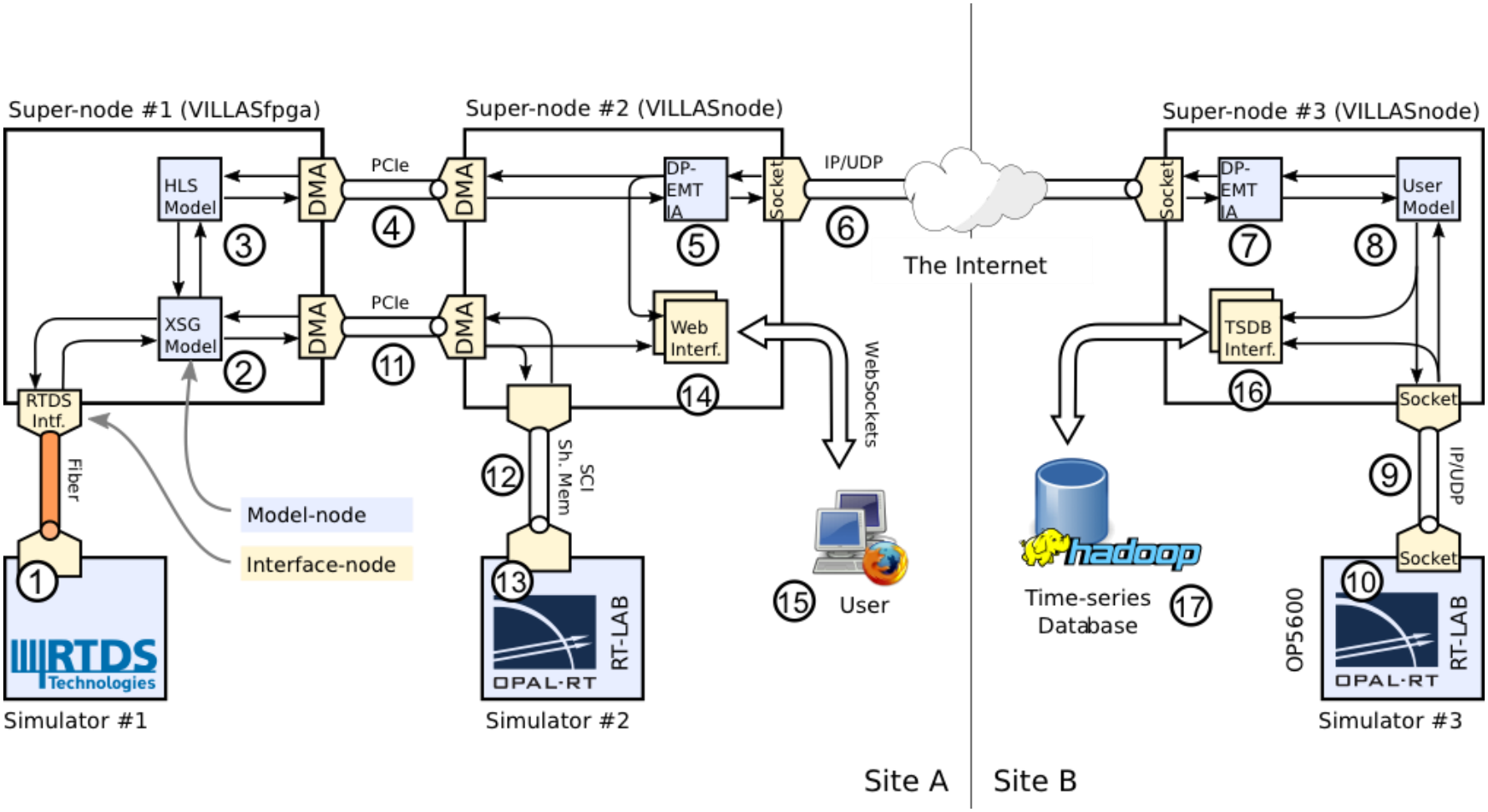
- General design objectives
  - A holistic framework with modular and generic architecture
  - Portability of interfaces among laboratories for integration of different local assets (digital real-time simulators, measurement devices, estimation and control algorithms)
  - Plug-and-play framework for geographically distributed test beds and co-simulation

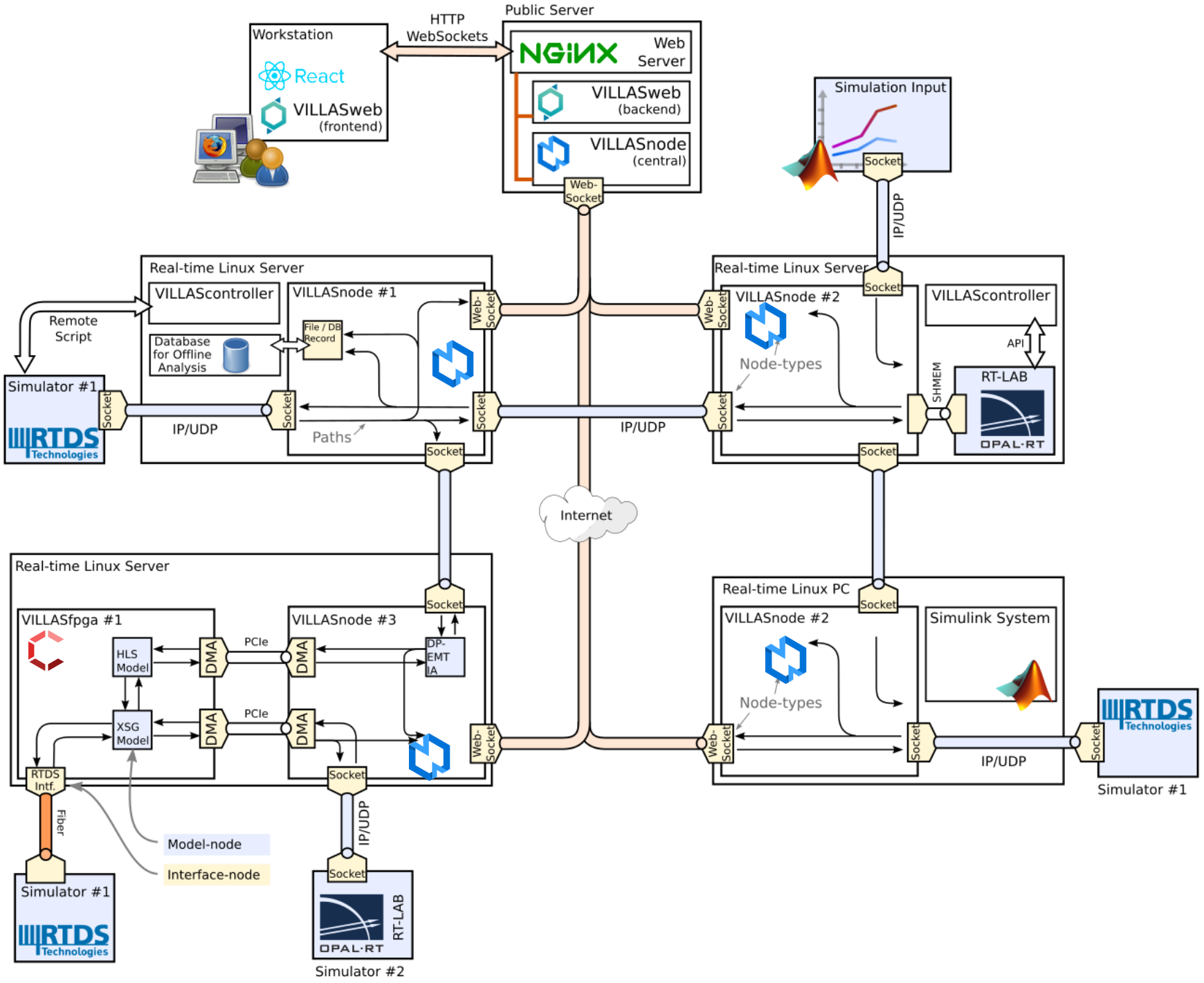
- The main pillars of VILLASframework
  - **VILLASnode** Gateway for connecting simulation equipement
  - **VILLASfpga** Extendended hard-realtime capabilties and FPGA-based models
  - **VILLASweb** Planning, Execution and Analysis of complex simulation scenarios
  - **VILLAScontroller** Unified API for controlling DRTS (OPAL, RTDS, Simulink) (planned)

VILLASframework
Overview



Complex topologies




Workstation
React
VILLASweb
(frontend)
HTTP
WebSockets
Public Server
NGINX
Web
Server
VILLASweb
(backend)
VILLASnode
(central)
Web-
Socket
Simulation Input
Socket
IP/UDP
Real-time Linux Server
VILLAScontroller
VILLASnode #1
Remote
Script
Database
for Offline
Analysis
File / DB
Record
Simulator #1
RTDS
Technologies
Socket
IP/UDP
Paths
Web-
Socket
Real-time Linux Server
VILLASnode #2
VILLAScontroller
Node-types
API
SHMEM
RT-LAB
OPAL-RT
IP/UDP
Internet
Real-time Linux Server
VILLASfpga #1
HLS
Model
XSG
Model
DMA
PCIe
VILLASnode #3
DP-
EMT
IA
RTDS
Intf.
Fiber
Simulator #1
Model-node
Interface-node
IP/UDP
Simulator #2
RT-LAB
Real-time Linux PC
VILLASnode #2
Simulink System
Node-types
IP/UDP
Simulator #1

VILLASnode

VILLASnode
Basics

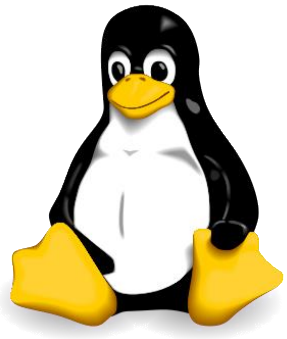
- Started as a project to connect internet distributed real-time simulators
  - Exchange of dynamic phasors over high-latency internet connections (> 20 ms RTT)
- Originally only for RTDS and OPAL
- Extensibility
  - Added compatibilty for more node-types later
  - It‘s a C framework / library & set of applications
    - = You can write your own and just use a subset of the functionality
- Running on Real-time Linux
  - Command line applications
  - Configuration via textfile (JSON-like)

VILLASnode
Design decisions

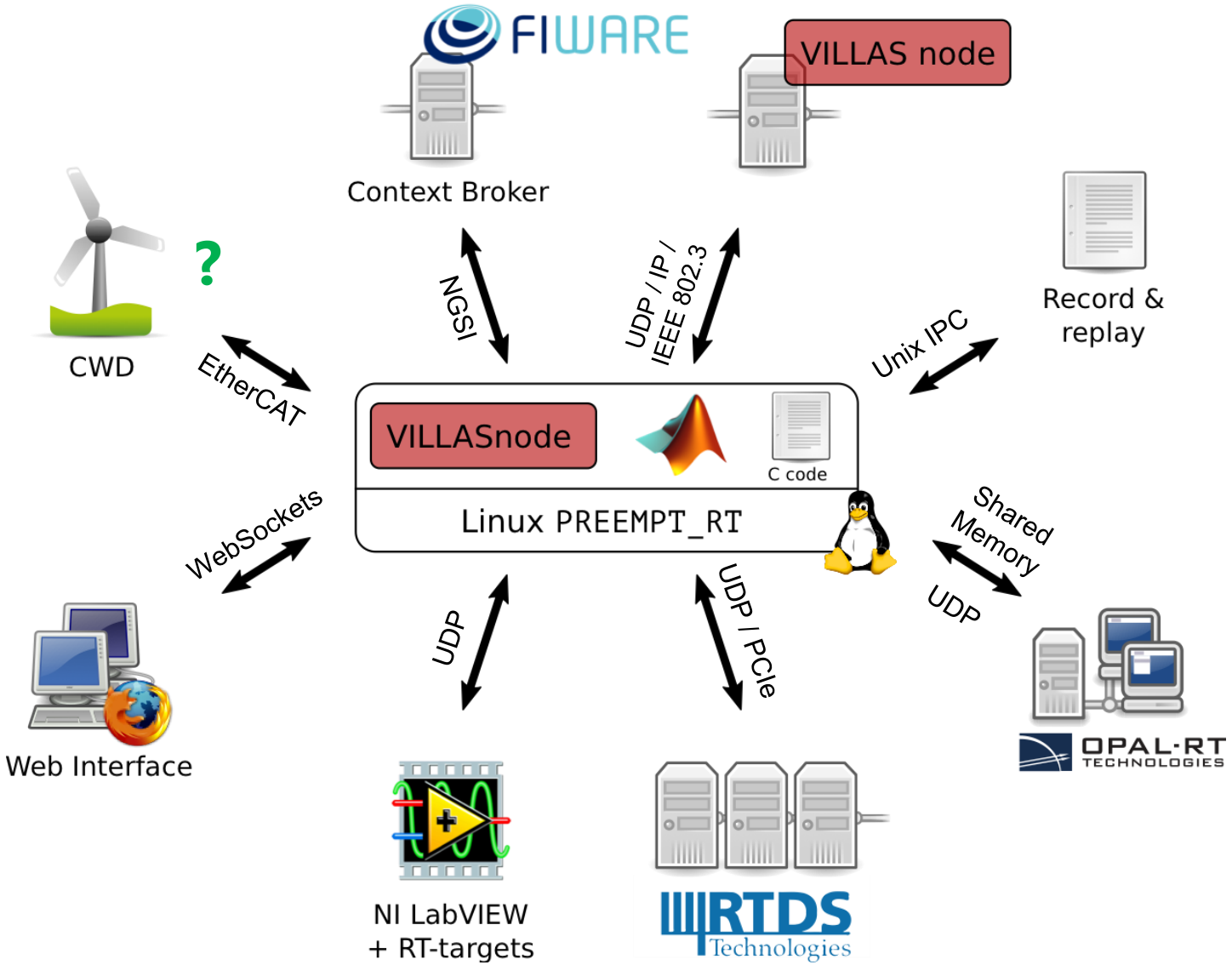
- Make interface as transparent to the model as possible
  - „Hide“ latency within line interface models or with dynamic phasors
  - ... or use network emulation to induce network effects artificially

Goals → Low & deterministic latency

- Object oriented low-level C for best performance
- Only depends on **open source** tools & libraries
- Make use of **Linux real-time** features (PREEMPT_RT patchset)
- Multi-threaded, non-blocking design
- Reserved CPU cores for execution

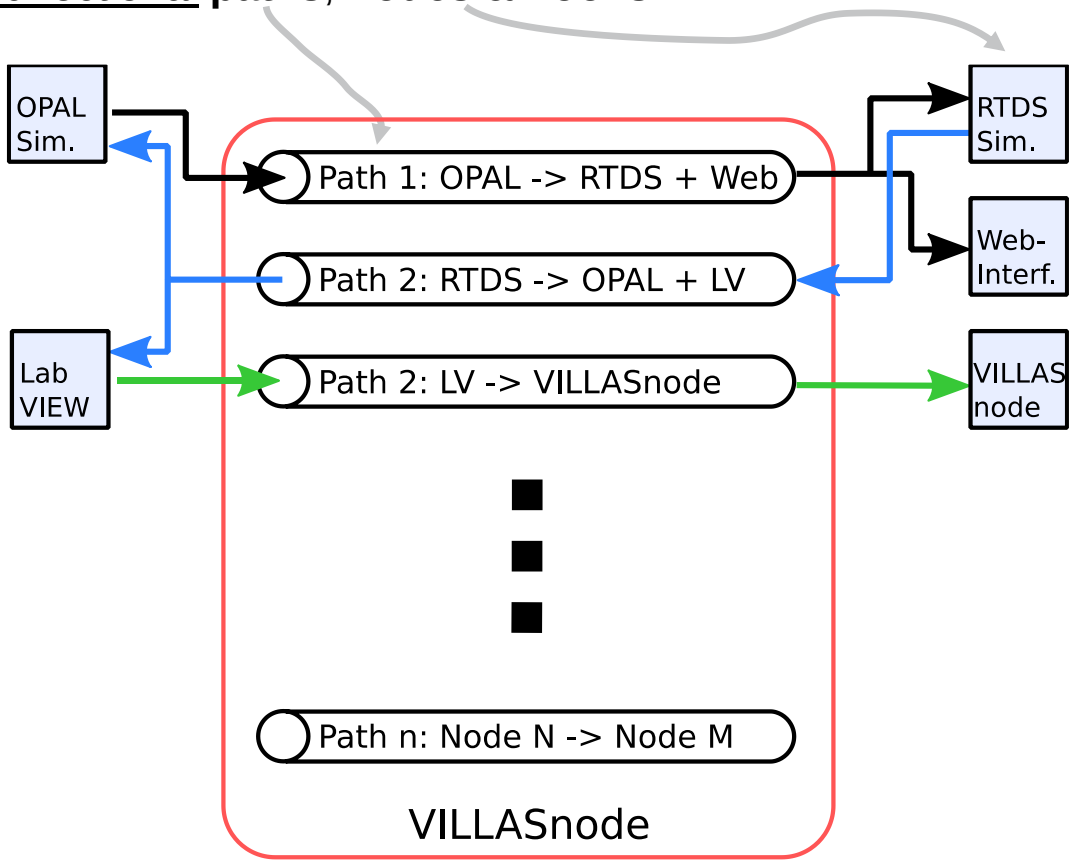


VILLASnode


FIWARE
Context Broker
VILLAS node
CWD
?
EtherCAT
NGSI
UDP / IP / IEEE 802.3
Unix IPC
Record & replay
VILLASnode
C code
Linux PREEMPT_RT
WebSockets
Shared Memory
UDP
UDP
UDP / PCIe
Web Interface
OPAL-RT TECHNOLOGIES
NI LabVIEW + RT-targets
RTDS Technologies

VILLASnode
Concept

- Router for sample / value based simulation data
  - **n-to-n** forwarding of sample values
  - MUX and DEMUX supported
  - Concept of unidirectional **paths**, **nodes** & **hooks**



VILLASnode
Concept

- Nodes
  - Representation of a Simulator / Model
  - Two types:
    - Model: runs on same machine
    - Interface: connect an external simulator
  - Acts as a source or sink of samples
- Paths
  - Unidirectional connection between nodes: 1-to-N (1 source, N sinks)
  - Hook functions
    - Interface algorithms
    - Dynamic phasor conversion (DFT)
    - Down-sampling
    - Timestamping
    - Collect statistics
- Hooks
  - Process / Filter the forwarded data with user-defined functions

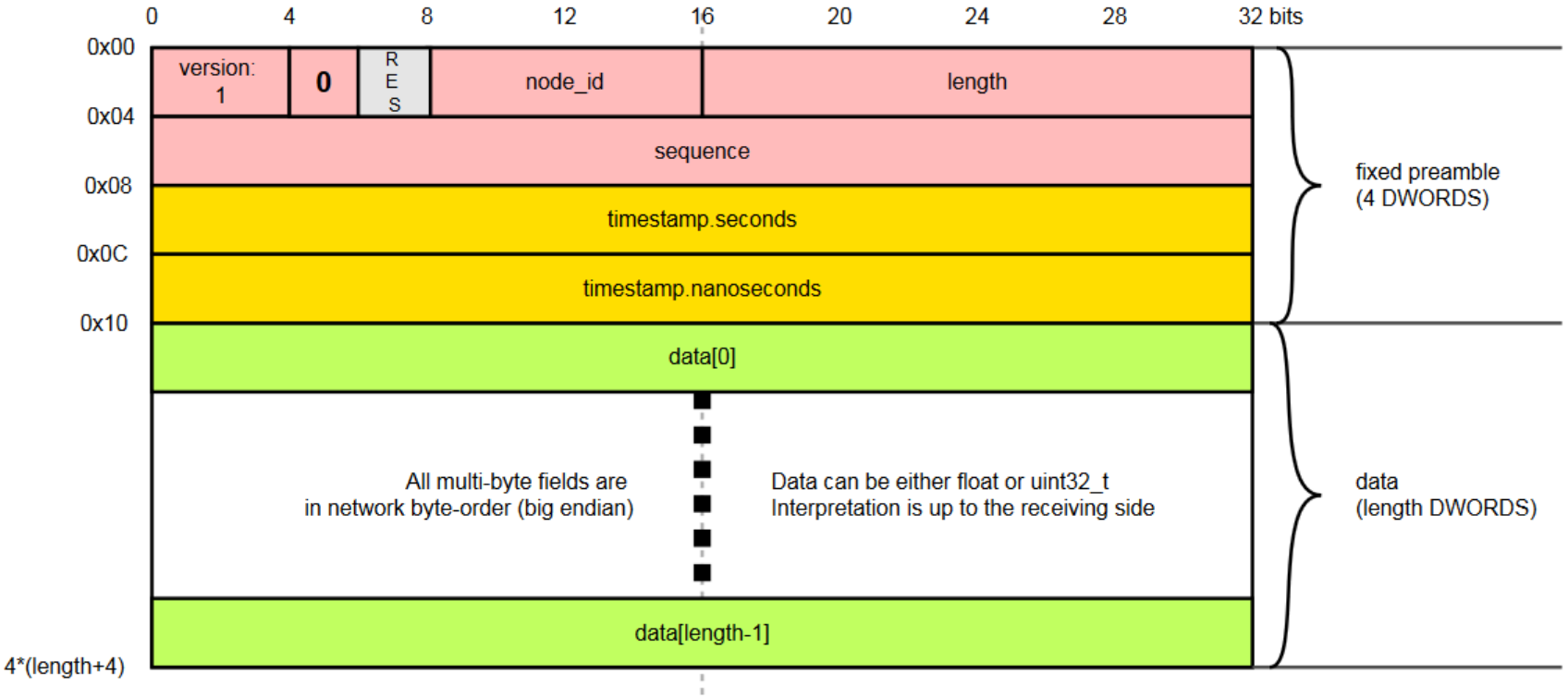
VILLASnode
Node-types

Node-type	Description	Note
`socket`	Standard BSD Sockets	UDP, IP, IEEE802.3 Support for RTDS‘ GTNETv2-SKT
`websocket`	Connection-oriented interface for webbrowsers	• Firefox Browser • VILLASweb
`ngsi`	Next Genereration Services Interface	FIWARE Orion Context Broker
`file`	Reading and logging from files on harddisk	• Static load profile • Replay
`opal`	OPAL-RT Shared Memory	Not synchronized (libOpalAsyncApi)
`fpga`	VILLASfpga	PCIe Interface + DMA to RTDS (GTFPGA)

VILLASnode
Node-types (cont‘d)

Node-type	Description	Note
`cbuilder`	Run RTDS Cbuilder components	(depracted)
`shmem`	Shared memory to processes running on the same machine	

VILLASnode
Network Packet Format for socket and websocket node-types



This protocol is only used by socket and websocket node-types!
Each node-type can define it‘s own protocol (e.g. NGSI uses HTTP REST + JSON)

VILLASnode
Features Overview

- Supports heterogenous environments
  - OPAL-RT
  - RTDS
  - LabVIEW
  - Custom Ethernet / IP / TCP & UDP Transports
- Synchronization
  - Timestamping
  - NTP / PTP
  - Fixed Rate Sending
- QoS Monitoring
- Network Emulation
- Flexible Hook Functions
  - Dropping of re-ordered UDP packets
  - FIR Filtering
  - Logging
- Tools
  - Generate random data
  - Manual send / receive
  - Replay
- Very low-latency forwarding
  - Thread / IRQ Pinning
  - CPU isolation
  - Zero-copy
  - Multithreaded

VILLASnode Tools

- Standard IO streams

- Main server: `villas node`
  - Runs one or more paths
  - Multi-threaded
  - Collects statistics
  - Filter / Process data with hook functions
- Signal generator: `villas signal`
  - Generate square, sine, ramp, triangle and random signals
  - Adjustable timestep, amplitude & frequency
- Read / write from / to stdin / stdout: `villas pipe`
  - Pipe to / from files
  - User input
  - Debugging
- Run hooks with stdin / stdout data `villas hook`

VILLASweb

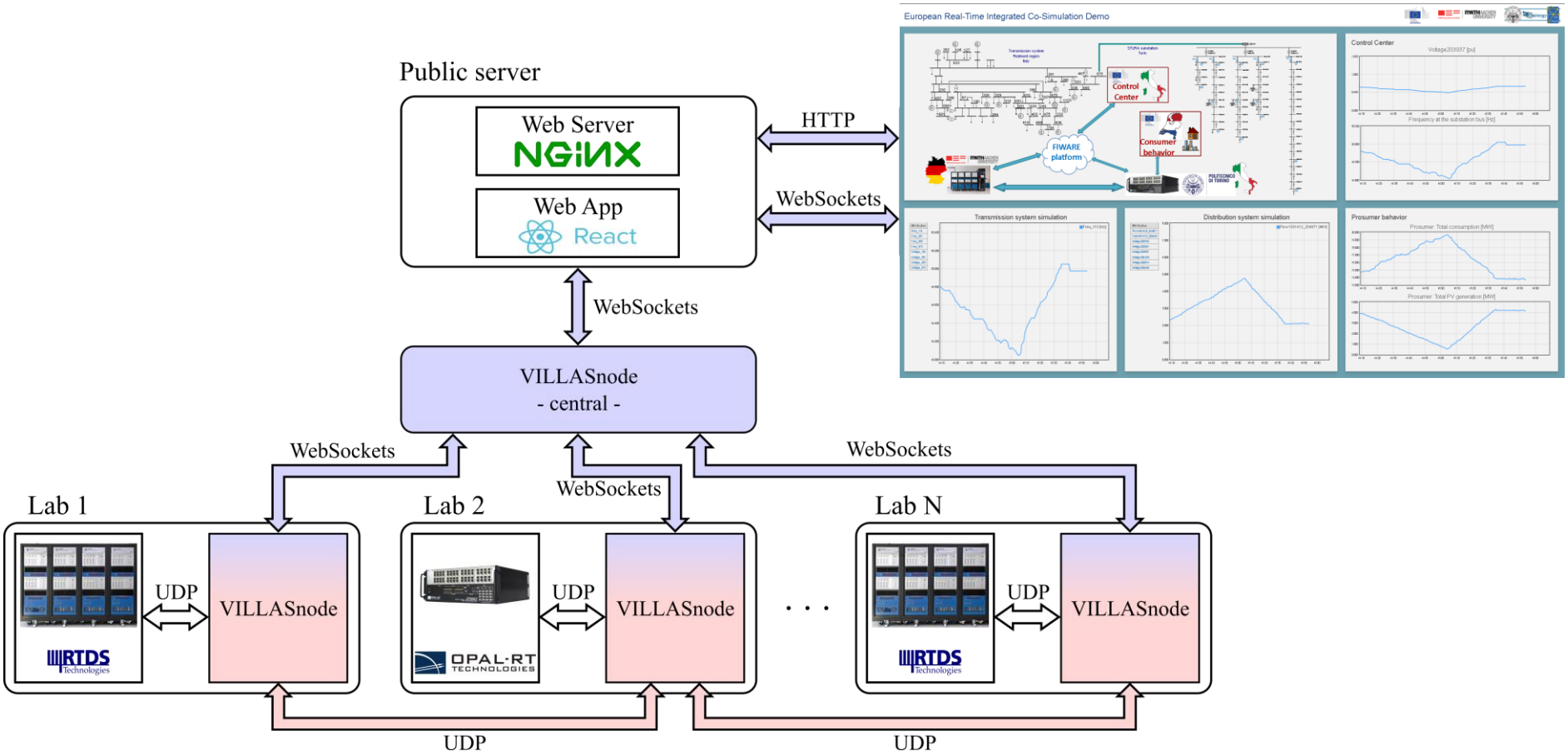
VILLASweb
Overview

- Web interface for planning, controlling & analysing the distributed simulation
- Work-in-Progress

- The stages
  1. Planning of simulation
     = Upload models
     = Upload input data
     = Map models onto simulators
  2. Running simuation
     = Live-stream of selected Voltages / Currents / etc.
     = User interaction with button / sliders
  3. Analysis / Collection of results
     = Download results to workstation

VILLASweb

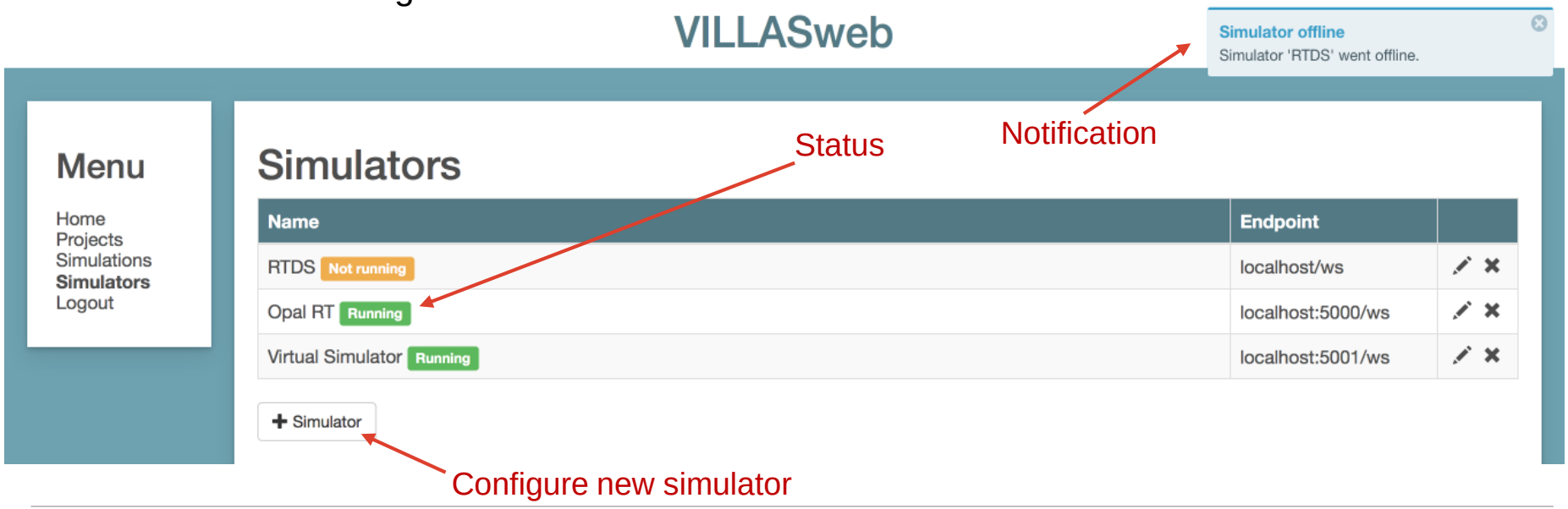
- Consolidated monitoring of the co-simulation with measurements from all subsystems



VILLASweb
Overview

- Identity Management

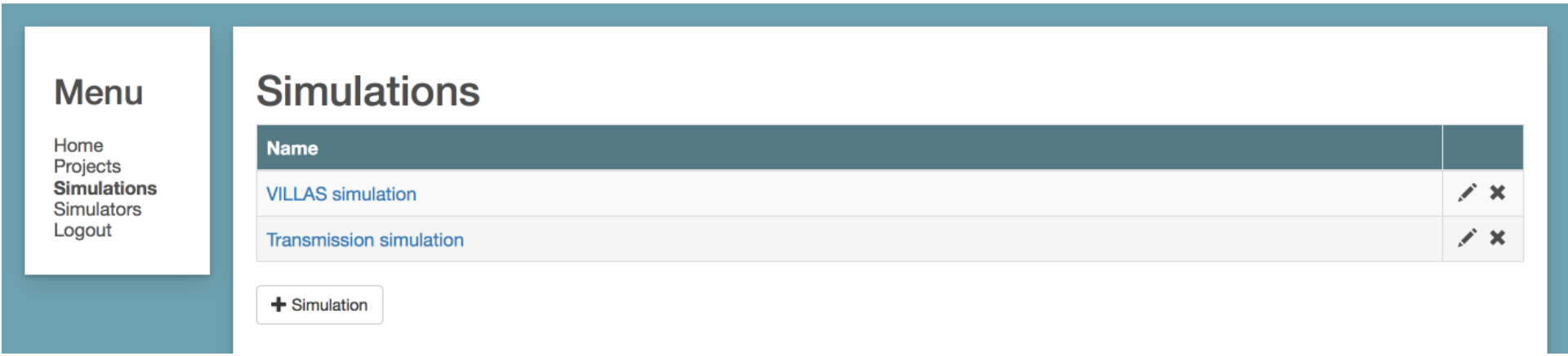
- Menu overview – Simulators
  - Status and configuration of simulators



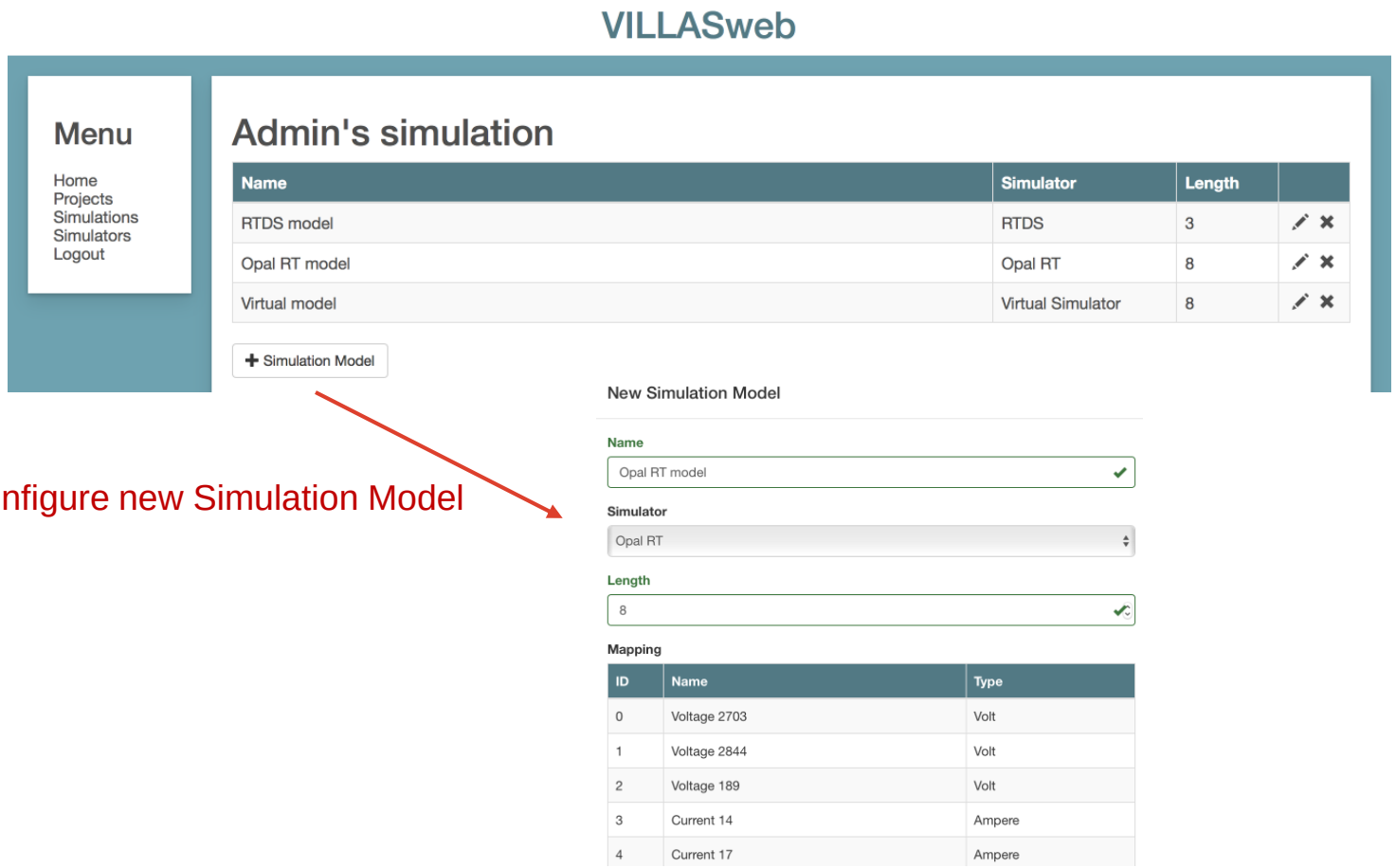
E.ON Energy Research Center | RWTH AACHEN UNIVERSITY

VILLASweb
Overview (cont'd)

- Menu overview – *Simulations*
  - Multiple *Simulations* can be created and configured for available simulators
  - A *Simulation* refers to a setup of a co-simulation experiment with defined subsystem models
    - For instance, we can create multiple *Simulations* for RT-Super Lab Demo that refer to different stages of demo development



- Menu overview – *Simulation Models*
  - *Simulation* configuration requires configuration of *Simulation Models* for simulators that are included in the co-simulation experiment



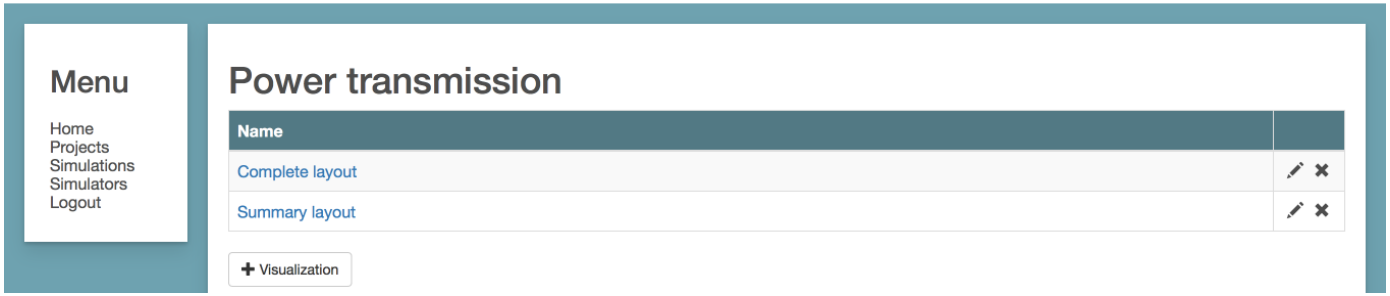
VILLASweb
Overview (cont'd)

- Menu overview – *Projects*
  - Multiple *Projects* can be created and configured for a *Simulation*
    - A *Project* can refer to a specific case study of a co-simulation experiment

VILLASweb

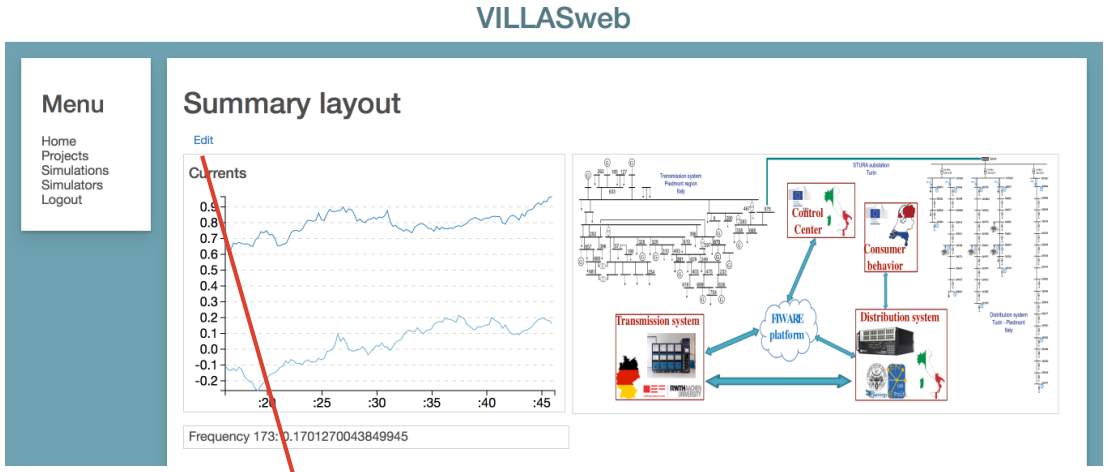
- Menu overview – *Visualizations*
  - Multiple *Visualizations* can be created and configured for a *Project*
    - A user can create different layouts (e.g. Summary or Detailed) for monitoring of simulation results

VILLASweb

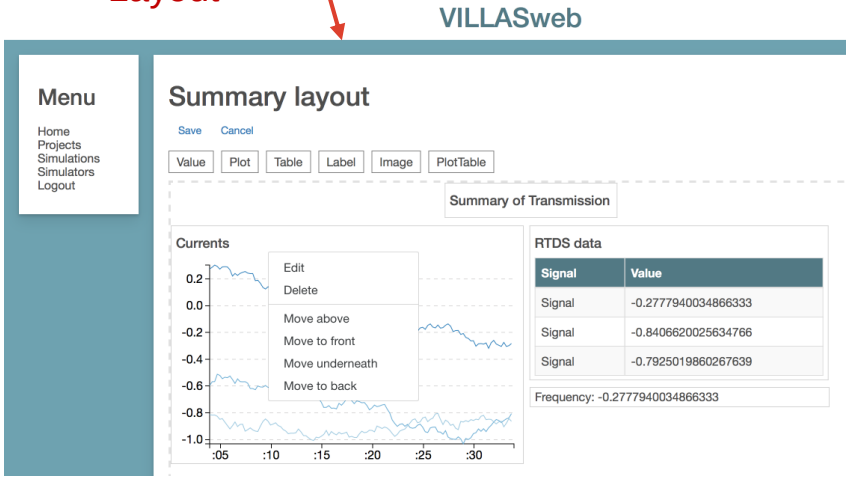


VILLASweb
Overview (cont'd)

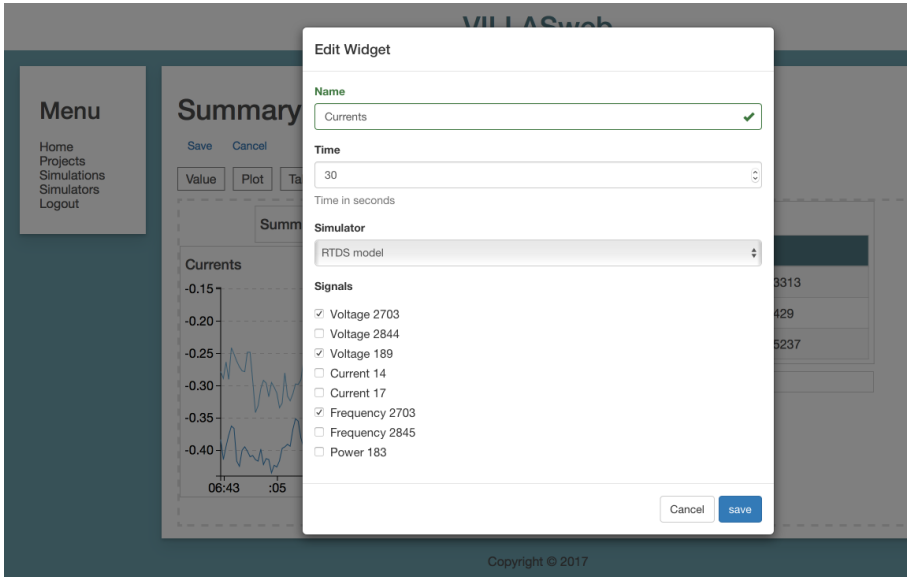
- A user can create a customized visualization of the co-simulation experiment



Edit Visualization Layout



Edit Plot Widget



E.ON Energy Research Center | RWTH AACHEN UNIVERSITY

VILLASfpga

VILLASfpga

- Extend VILLASnode instances with FPGA ressources
  - Interface DRTS
    - RTDS via GTFPGA netlist
    - Typhoon, OPAL-RT via Aurora protocol
  - Run models / interface-algorithms
    - Simulink / Xilinx System Generator models
    - C++ code using High Level Synthesis

VILLAScontroller

VILLAScontroller

- Provide a unified API for controlling a heterogenous environment of simulators
  - Load Model
  - Set Parameters
  - Start / Stop Simulation
  - Retrieve status of simulator

- Used by VILLASweb

- Planned for Q3/Q4 2017
  - Implemented in NodeJS

Deployment

Requirements: VILLASnode & Tinc

- Network
  - Preferably point-to-point GigEthernet connection to OPAL-RT / RTDS
  - Separate Ethernet connection to the Internet (may be in DMZ)
- Commodity PC / Server
  - Intel **x86-64** architecture
  - Bare metal Linux-based operating system
    - No background processes & GUI
  - RT_PREEMPT Kernel Patch set
  - Nice to have:
    - Dual port Network Interface Card (NIC)
    - Support for IEEE-1588: Precision-Time-Protocol (PTP)
    - Recent Multicore CPU (for best determinism)
- Use OPAL-RT target as host?
  - Already fulfills most of the requirements above ☺
- Linux knowledge is helpful

Deployment Procedure

Multiple options:

1. We provide you a ready-to-go Live Image (.iso)
   - ≡ Which can be booted on most commodity PCs / servers
   - ≡ Can establish tunnel to ACS for remote administration
     - = Allows us to do the final setup remotely

2. We provide you with a shell script which gives us remote access to your OPAL-RT simulator
   - ≡ We will install Tinc and VILLASnode on your OPAL-RT system

3. We provide you with instructions and support to setup the system yourself
   1. You can build it from source
   2. You can use available RPM packages for Fedora, Redhat, CentOS Linux

4. (Docker images **only** for testing)

VILLASframework
Materials

- Project website: https://villas.fein-aachen.org/website/framework/
  - Documentation: https://villas.fein-aachen.org/doc
  - Source code: https://git.rwth-aachen.de/VILLASframework
  - Docker images: https://hub.docker.com/u/villas/
  - ISO images: https://villas.fein-aachen.org/iso

GitLab

VPN networking / Security

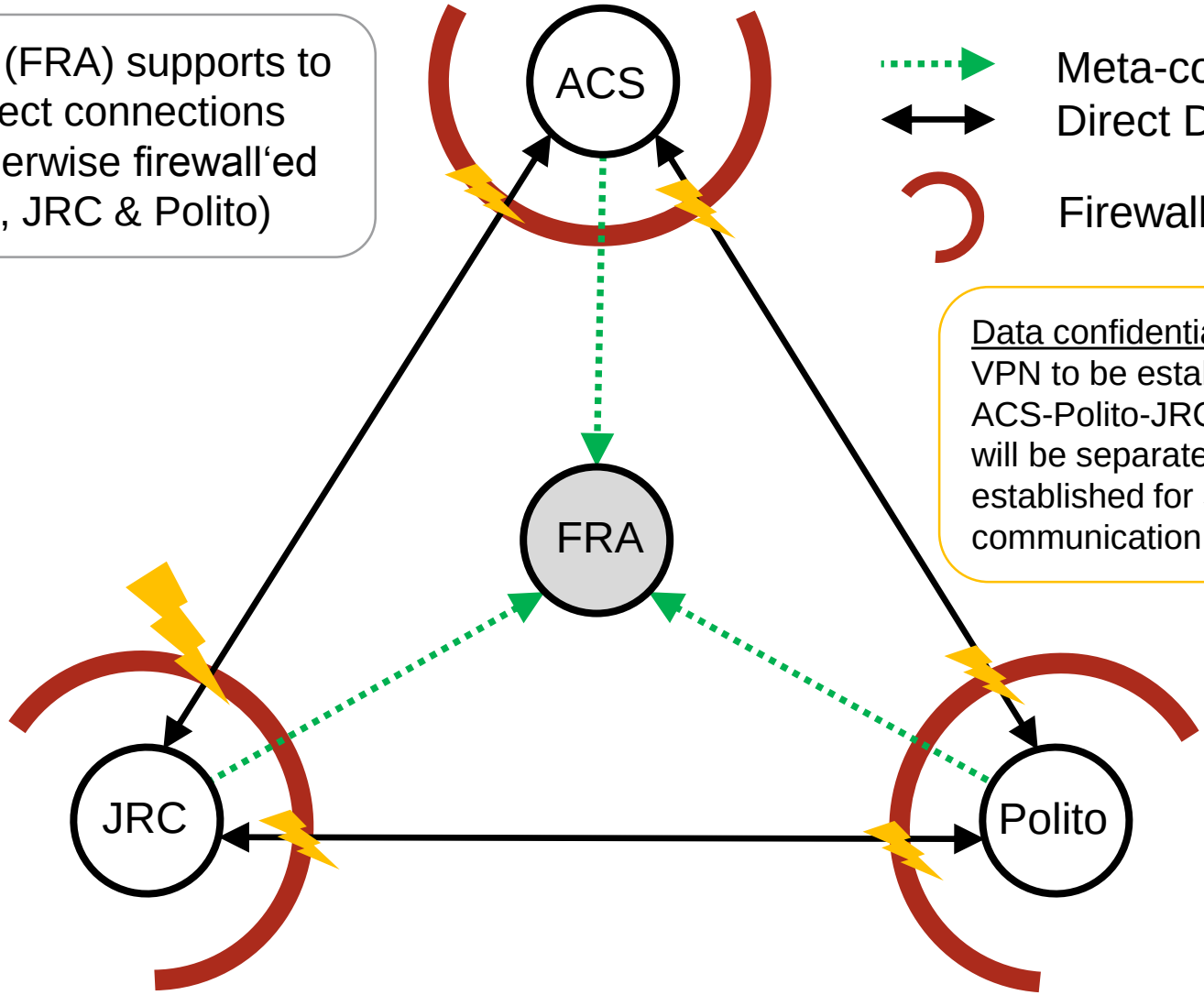
Tinc VPN

- Actively developed Open Source software
- Layer 2 or 3 VPN (Ethernet or IP)
  - Linux tap or tun device
- Tunnels data *over* UDP
- Fully decentralized + meshed
- Optional Encryption + Compression
- Enables to circumvent firewalls!
  - NAT hole punching
  - Comparable to Skype & Teamviewer (STUN)

http://www.tinc-vpn.org

Tinc VPN: Principle & Topology

Public node (FRA) supports to establish direct connections between otherwise firewall'ed nodes (ACS, JRC & Polito)



Meta-connection

Direct Data-path

Firewall

Data confidentiality aspects: VPN to be established for ACS-Polito-JRC connection will be separated from VPN established for SINTEF-ACS communication

Contact

E.ON Energy Research Center
Mathieustraße 10
52074 Aachen
Germany

Steffen Vogel, M.Sc.
T +49 241 80 49703
F +49 241 80 49709
stvogel@eonerc.rwth-aachen.de
http://www.eonerc.rwth-aachen.de

ACS | Automation of Complex Power Systems

E.ON Energy Research Center

RWTH AACHEN UNIVERSITY

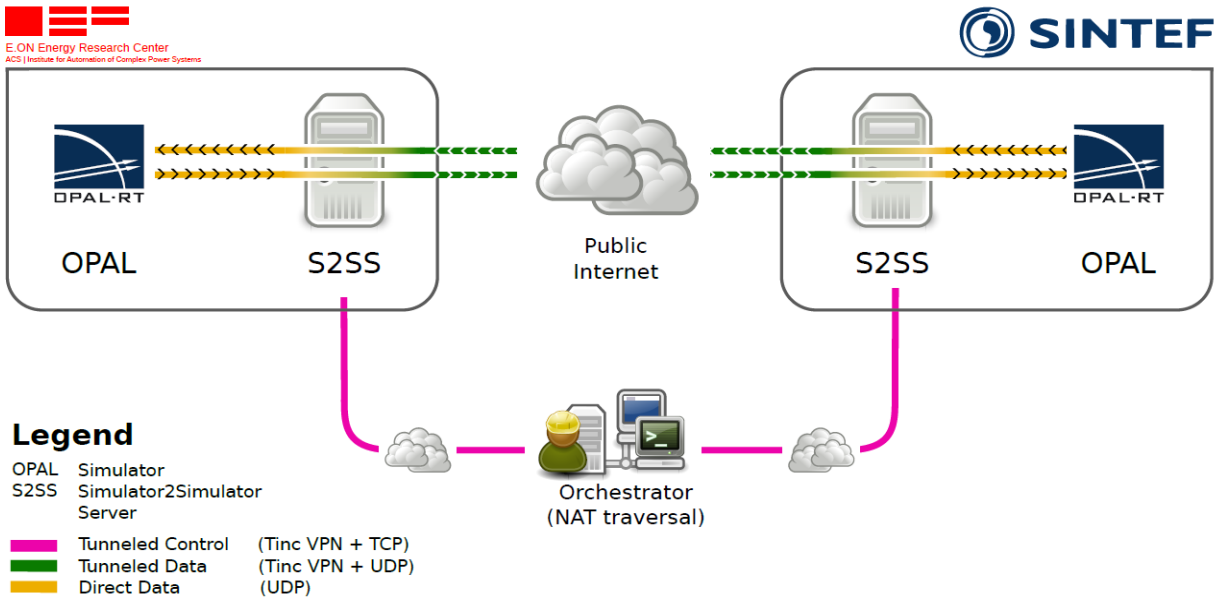
Current & Past Users

- RT-SuperLab (8 PoP)
  - Idaho National Lab (INL), USA
  - National Renewable Energy Laboratory (NREL), USA
  - Sandia National Laboratory (SNL), USA
  - Colorado State University (CSU), USA
  - Center for Advanced Power Systems (CAPS)
    Florida State University (FSU), USA
  - University of South Carolina (USC), USA
  - Polytechnic University of Turin (Polito), Italy
  - Institute for Automation of Complex Power Systems (ACS)
    RWTH Aachen University, Germany
- RE-SERVE (4 PoP)
  - Universitatea Politehnica Din Bucuresti (UPB), Romania
  - University College Dublin (UCD), Ireland
  - Polytechnic University of Turin (Polito), Italy
  - Institute for Automation of Complex Power Systems (ACS)
    RWTH Aachen University, Germany
- ERIC Lab Demonstration
  - EU Joint Research Center, Petten, Netherlands
  - Polytechnic University of Turin (Polito), Italy
  - Institute for Automation of Complex Power Systems (ACS)
    RWTH Aachen University, Germany
- Pro-of-Grids / MARINET (2 PoP)
  - SINTEF NTNU Trondheim, Norway
  - Institute for Automation of Complex Power Systems (ACS)
    RWTH Aachen University, Germany
- University South Carolina (2 PoP)
  - Andrea Benigni
- ACS Spin-off: Gridhound
- Wideband System Identification
  - Master Thesis: Eyke Liekmann

PoP: Points-of-Presence

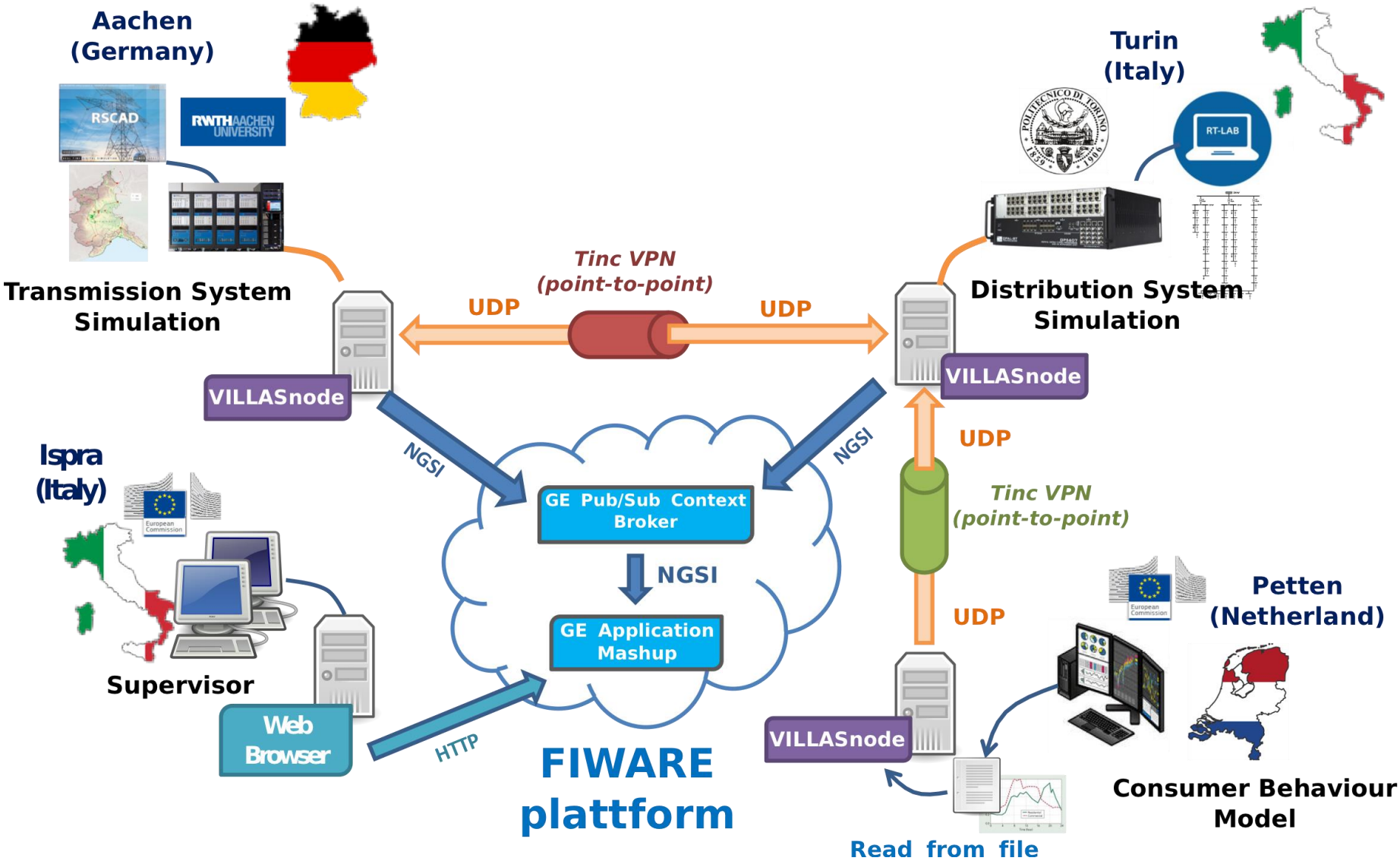
Internet-distributed simulation platform SINTEF-ACS

- Internet-distributed simulation platform connects two real-time digital simulators located at SINTEF (OPAL-RT) and ACS (OPAL-RT/RTDS)
  - Simulator-to-simulator Server (S2SS) PC in each laboratory manages data exchange with local simulator and Internet communication with remote server
  - VPN established for data security (Tinc – an open source VPN daemon)
    - Decentralized solution: central server not needed



E.ON Energy Research Center | RWTH AACHEN UNIVERSITY

ERIC Lab Demo



E.ON Energy Research Center | RWTH AACHEN UNIVERSITY

External Interconnections

Sintef, Norway

Joint Research Center, EU

Uni. South Carolina, US

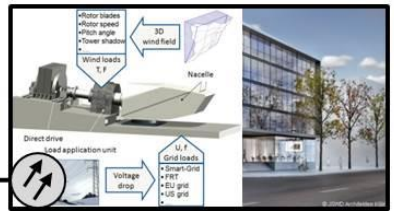
Politecnico di Torino, Italy

ACS Real Time Lab

5 MW PGS Test Bench

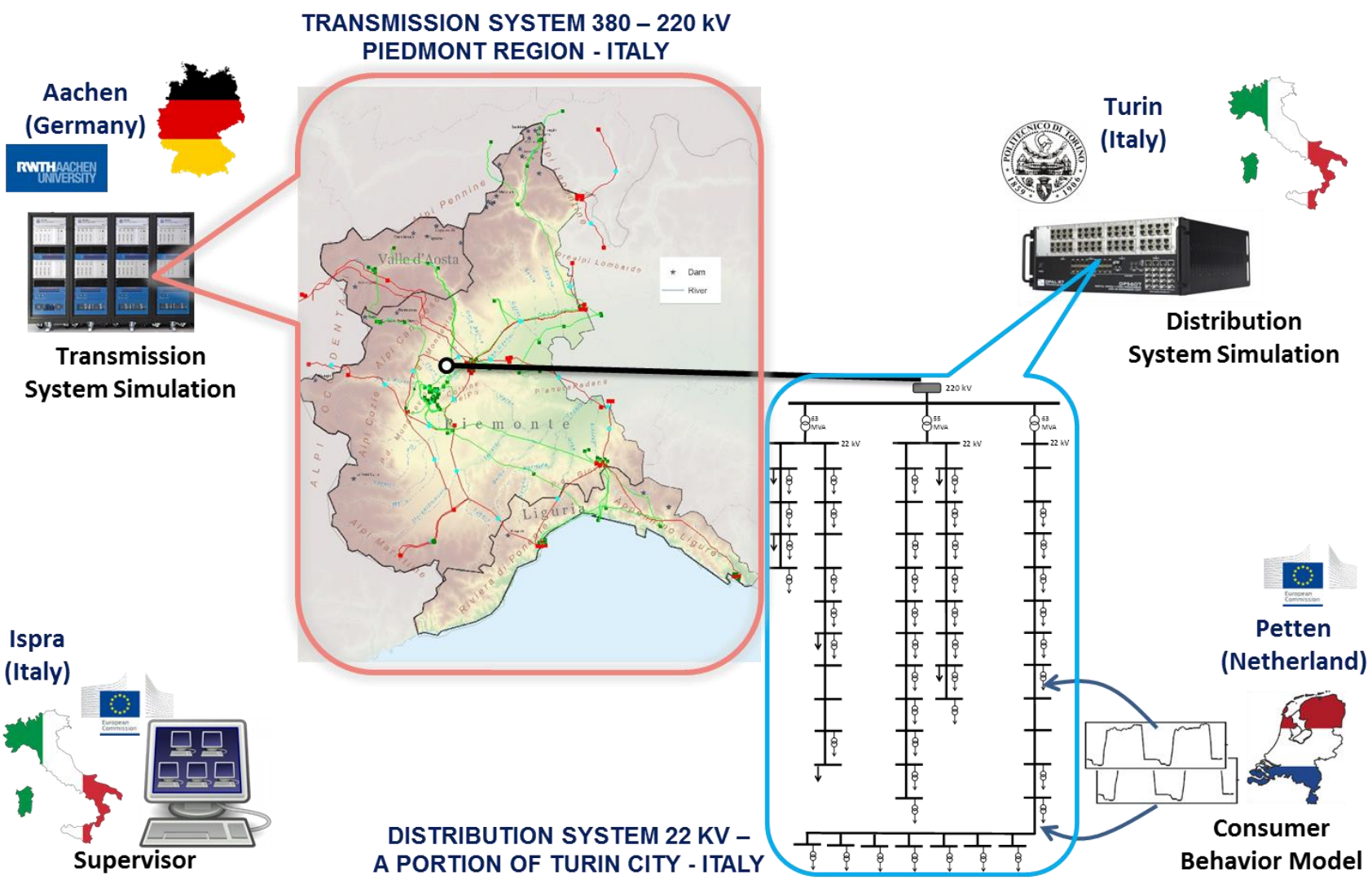
1 MW On-Shore Wind Turbine Test Bench

4 MW On-Shore Wind Turbine Test Bench



E.ON Energy Research Center | RWTH AACHEN UNIVERSITY

Example of virtual integration over Wide Area Network
ERIC Lab demonstration



E.ON Energy Research Center | RWTH AACHEN UNIVERSITY