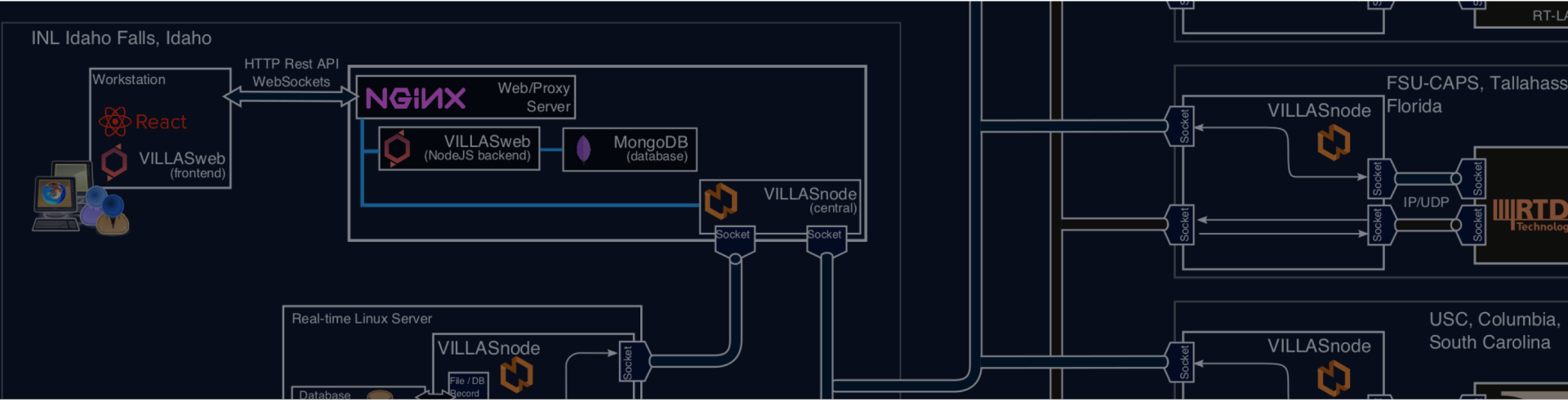


VILLASframework

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

A technical description of the framework, components and workflows
for the Technical Workshop of the ERIGrid Project

13.09.2018
Oldenburg, Germany

This project has received funding from the European Union's Horizon 2020 Framework Programme for Research and Innovation under grant agreement no 727481

Steffen Vogel, M.Sc.
(v5 – 13.09.2018)

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

Contents

- Design rationale
- Related Projects
  - Pintura
  - CIM++
  - DPsim
- VILLASframework
  - VILLASnode
  - VILLASweb
  - VILLAScontroller
  - VILLASfpga
- Future Goals

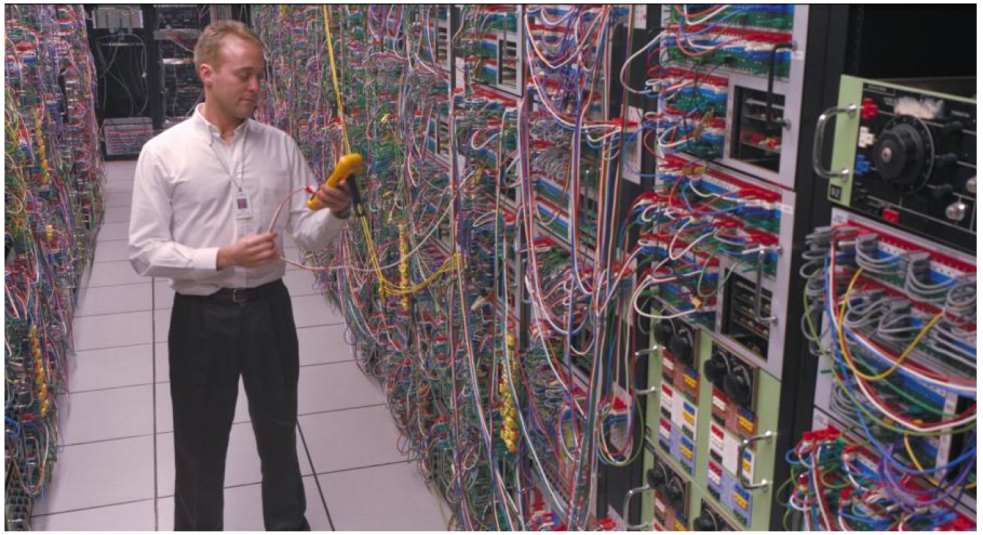
VILLAS
Framework for virtual integration of laboratories for co-simulation

- 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
  - for distributed power-system simulation and HiL testing
    - Less emulation of automation / SCADA / communication networks
  - 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
    - Simulation as a Service
    - Data as a Service
    - flexible access for third parties to leverage utilization of the infrastructure

Core Design Ideas

- A circuit switched network of real-time simulators
  - Point-to-point Lab Interconnections for optimal round-trip times
  - No central entity in control
- Lightweight protocols
- Separation of Control and Data Plane
- Designed for Real-time Simulations / HiL
  - EMT / Dynamic Phasors

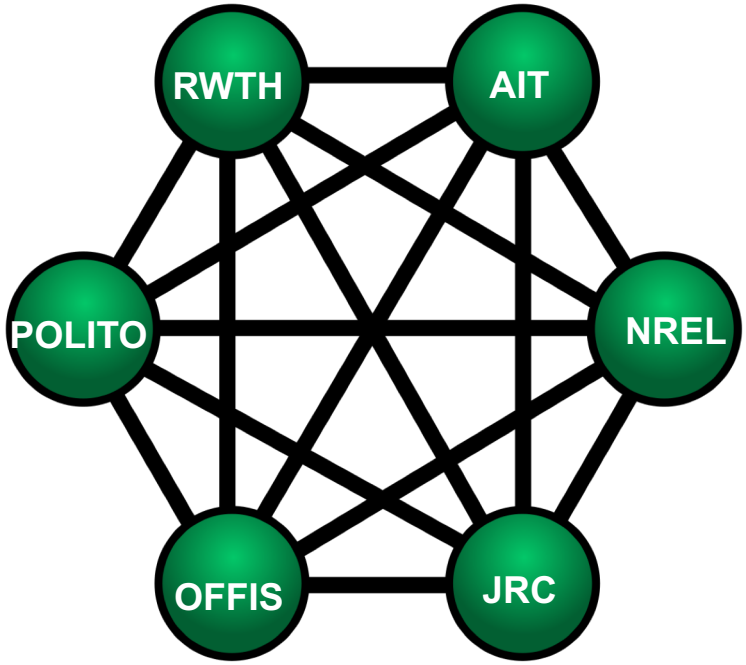


Tinc VPN

- Open Source software
- Layer 2 or 3 VPN
  - Linux tap or tun device
- Tunnels data *over* UDP
- Fully decentralized + meshed
- Optional Encryption + Compression
- Enables to circumvent firewalls
  - NAT traversal
  - Comparable to Skype & Teamviewer

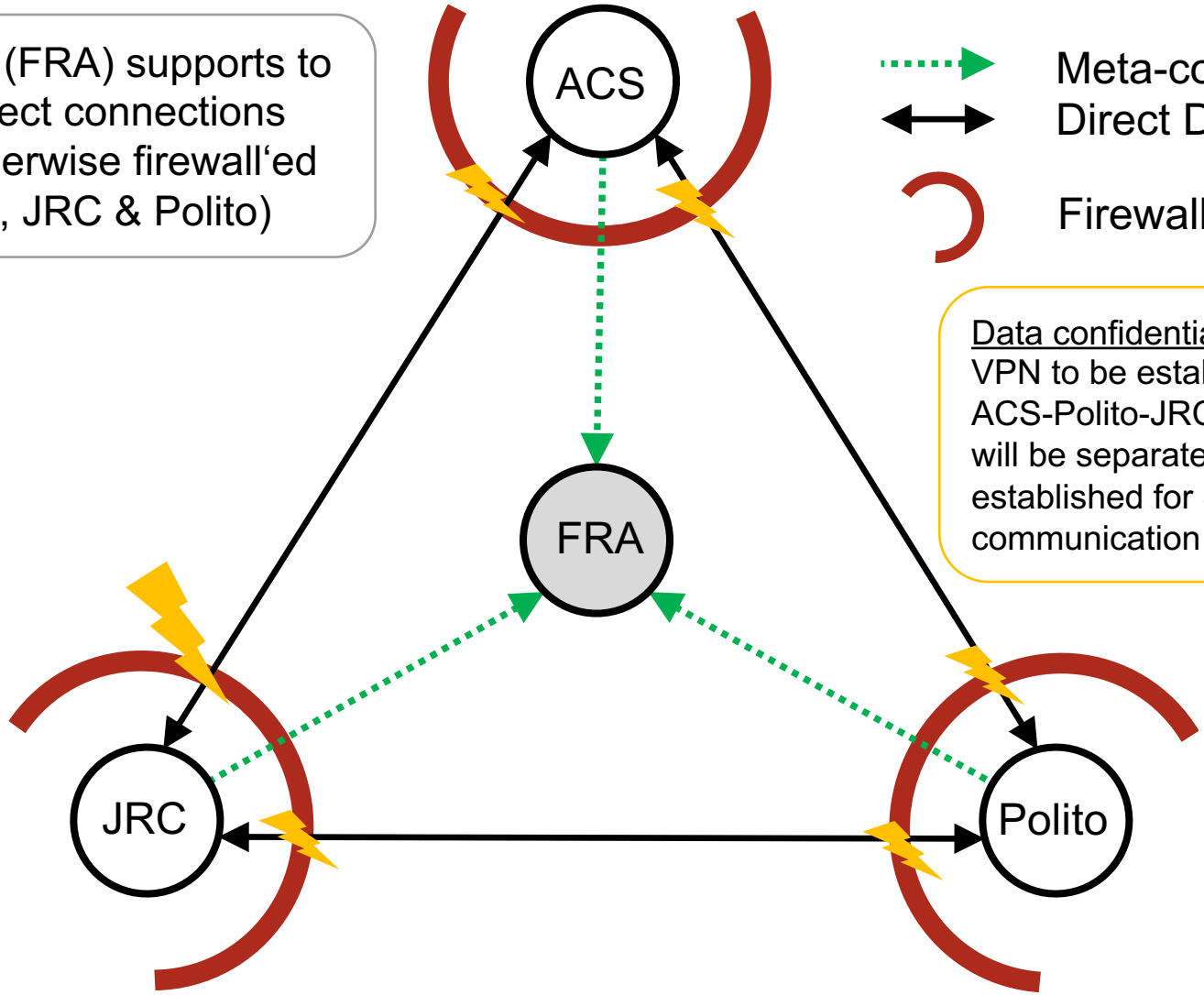
tinc

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)

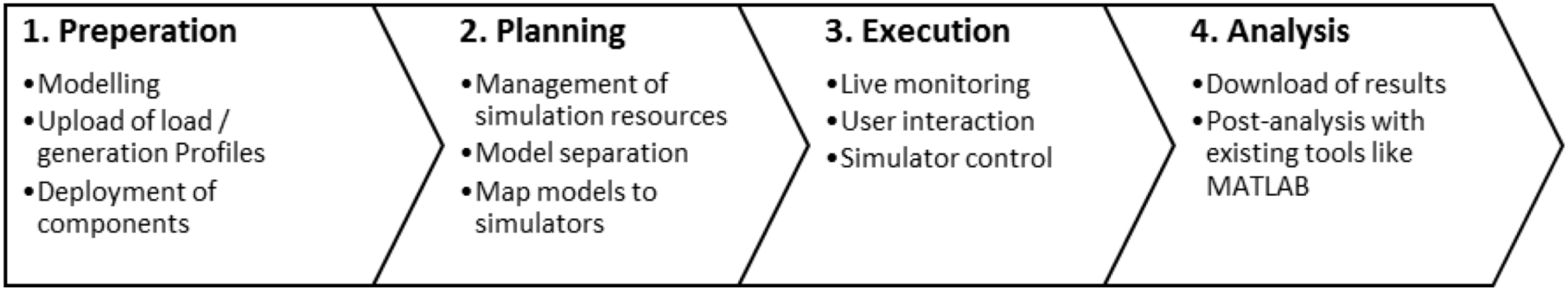


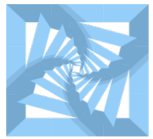
Meta-connection
Direct Data-path
Firewall

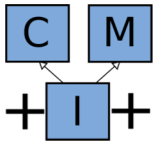
Data confidentiality aspects: VPN to be established for ACS-Polito-JRC connection will be separated from VPN established for SINTEF-ACS communication

VILLASframework
Workflow

- Formalization of a Workflow
- Guided by VILLASweb Interface







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 equipment
  - **VILLASfpga** Extended hard-realtime capabilities and FPGA-based models
  - **VILLASweb** Planning, Execution and Analysis of complex simulation scenarios
  - **VILLAScontroller** Unified RPC API for controlling DRTS & Equipment

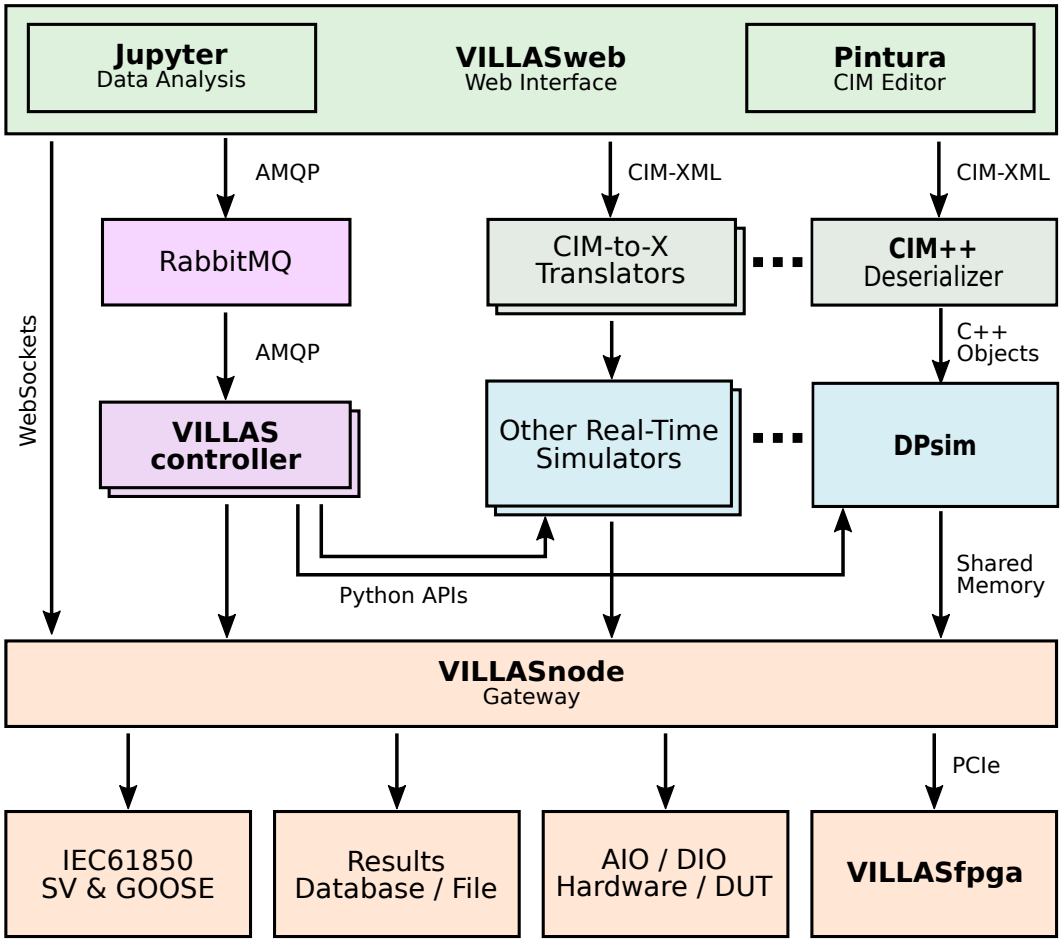
The Stack

User Interfaces

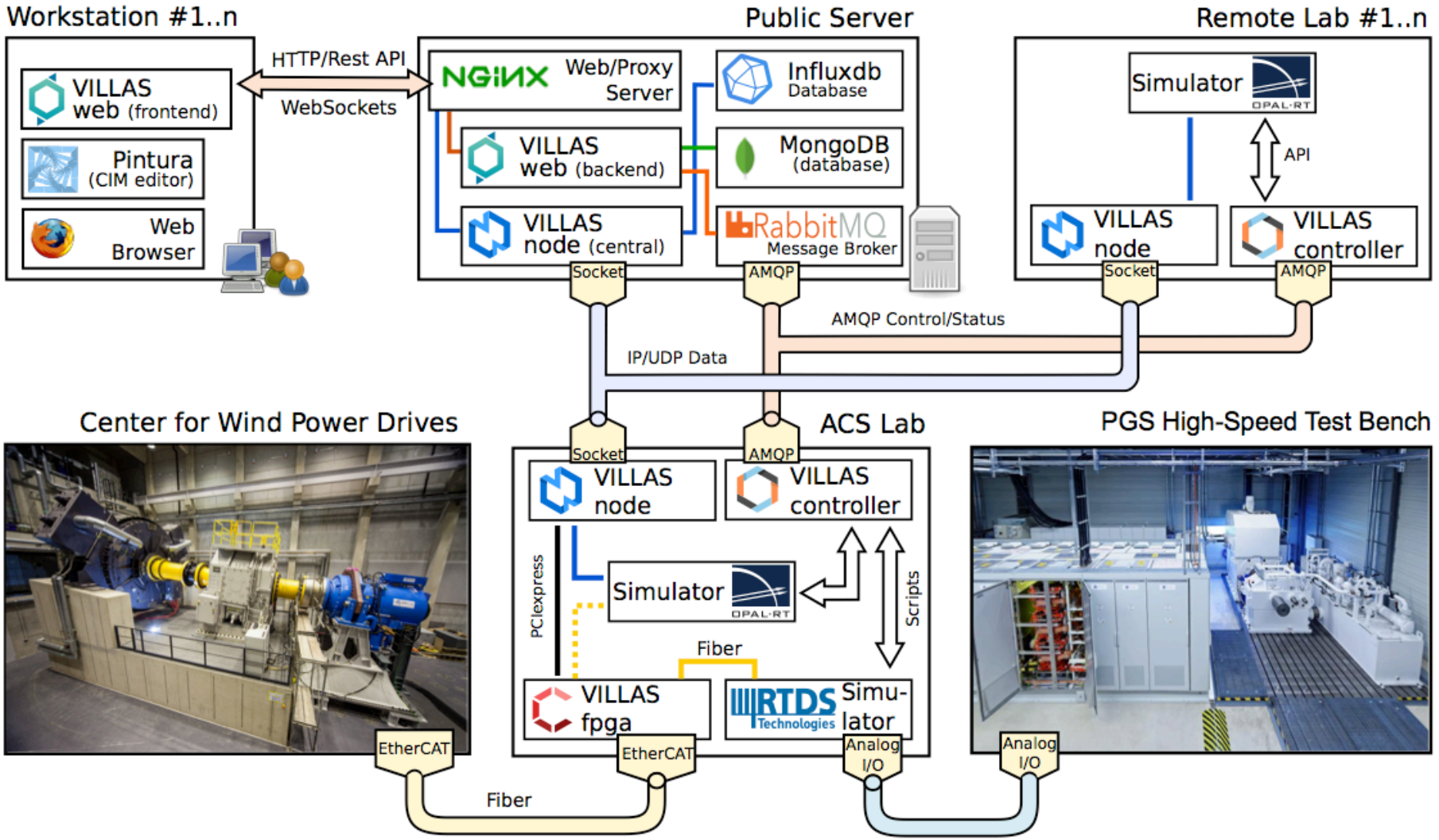
Simulators

Gateways

Interfaces

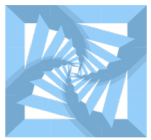


VILLASframework
Example Setup



CIM: Common Information Model
HTTP: Hypertext Transfer Protocol
API: Application Programming Interface

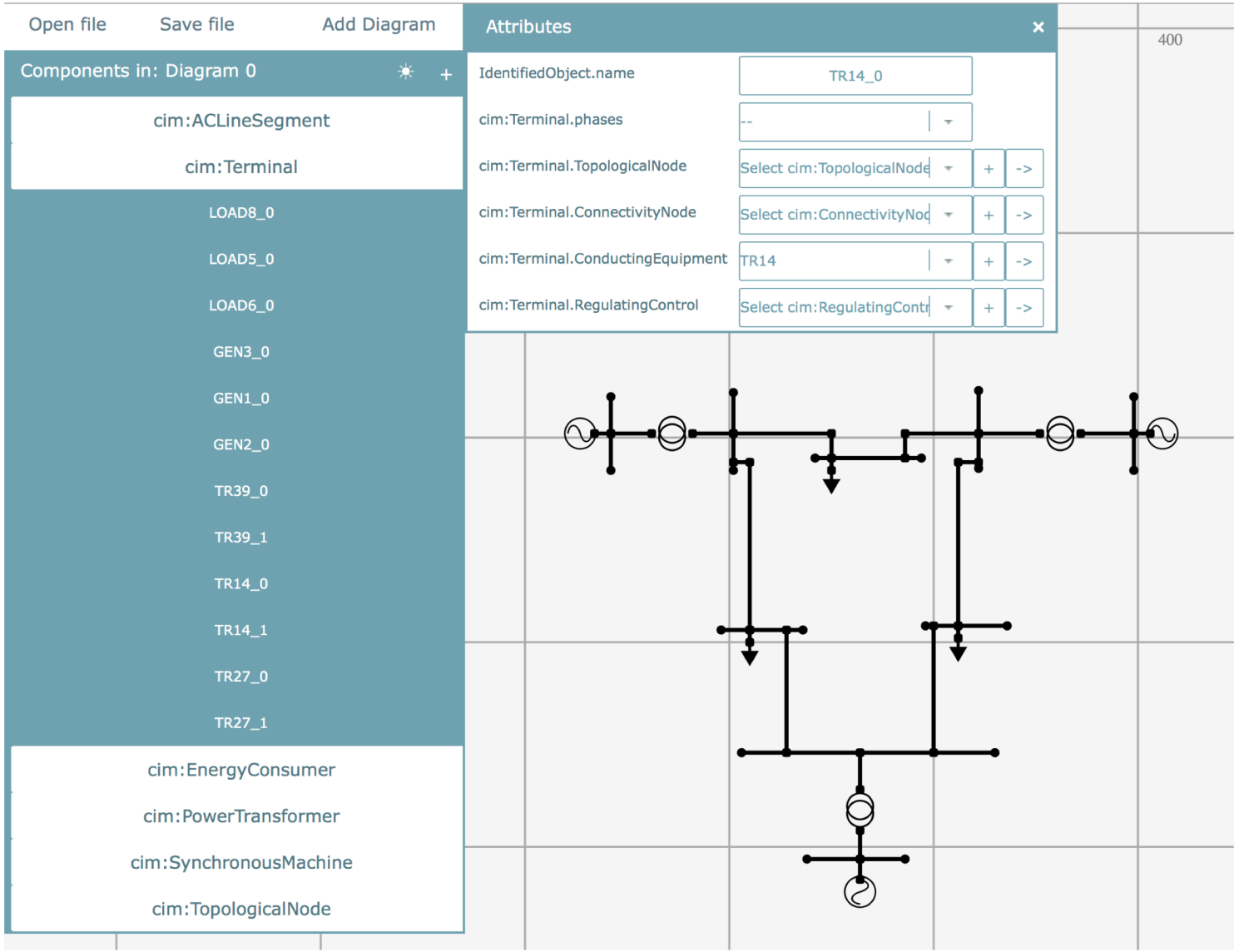
IP: Internet Protocol
ACS: Institute for Automation of Complex Power Systems
PGS: Institute for Power Generation and Storage Systems



Pintura

A web-based Editor for Common Information Models

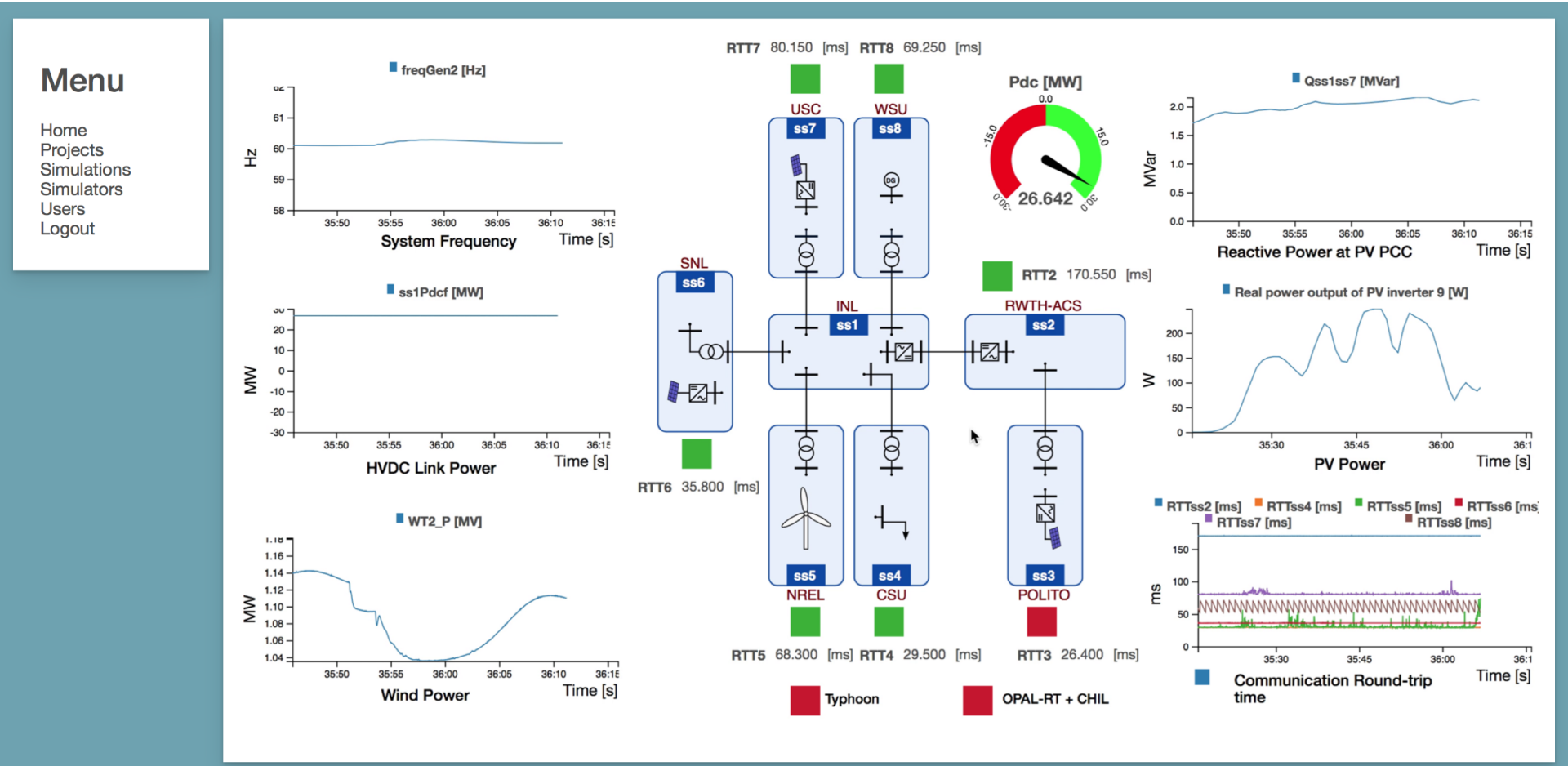
Pintura Screenshot



VILLASweb

An interactive web-interface for
managing, executing and monitoring large-scale co-simulations

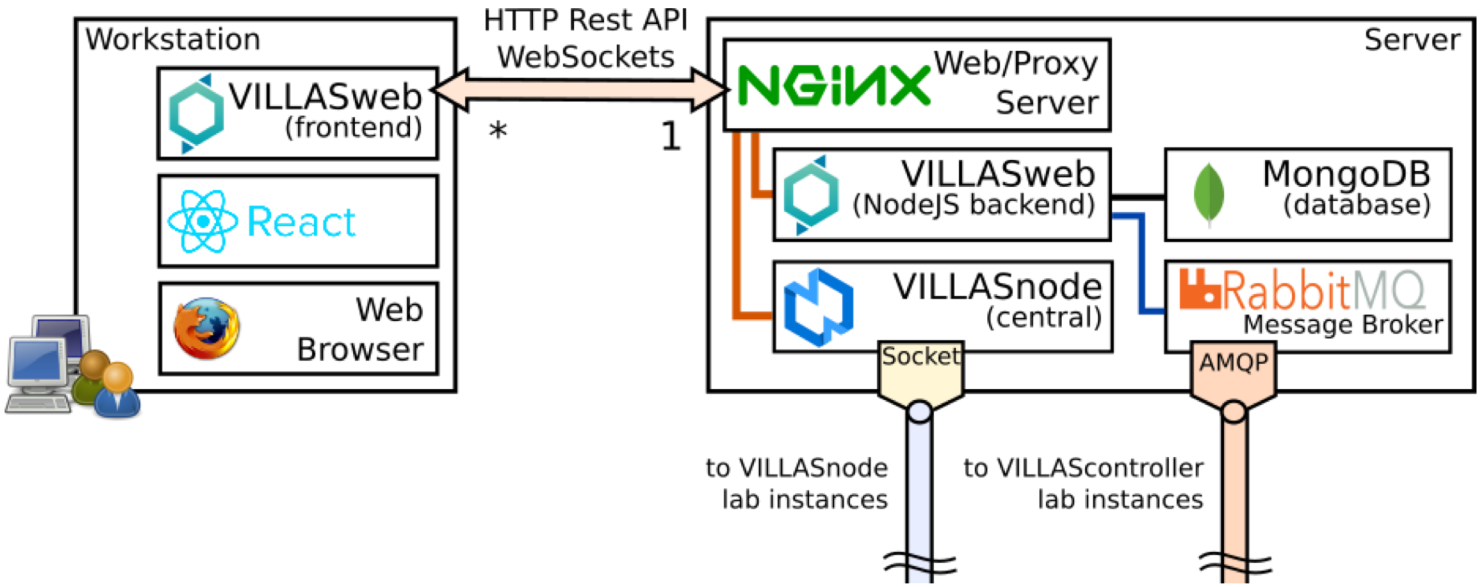
VILLASweb Screenshot



E.ON Energy Research Center | RWTH AACHEN UNIVERSITY

VILLASweb Architecture

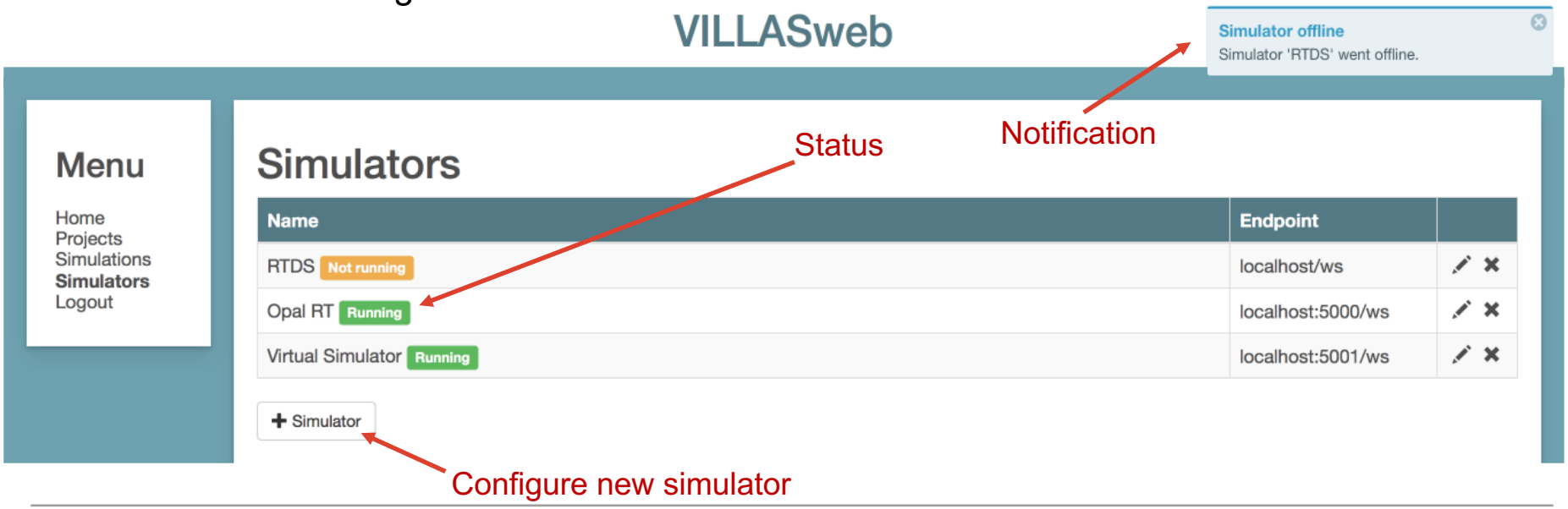
- VILLASnode for streaming live data
- VILLAScontroller / RabbitMQ for simulator control
- Nginx as load balancing proxy



VILLASweb
Overview

- Identity Management

- Menu overview – Simulators
  - Status and configuration of simulators

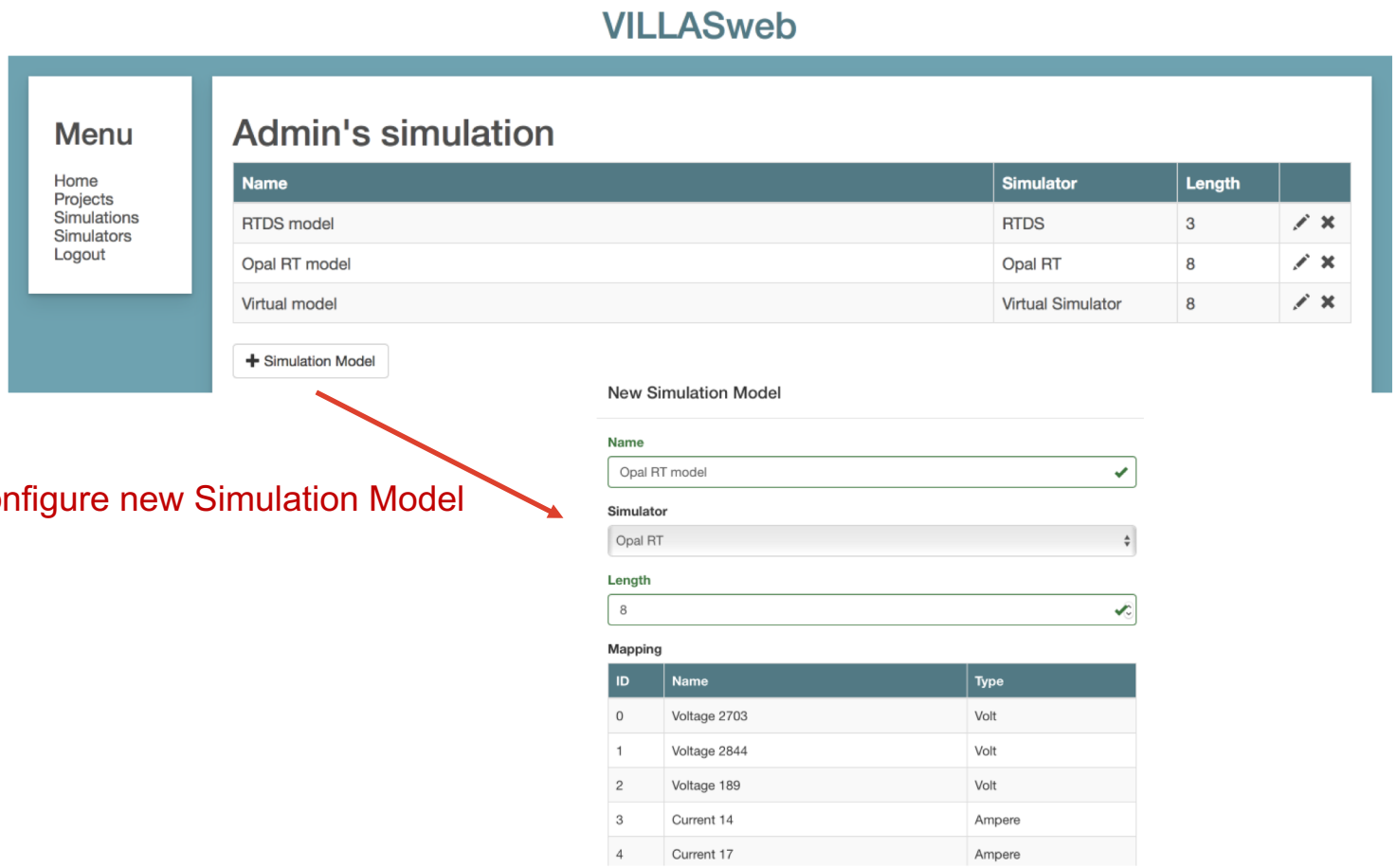


E.ON Energy Research Center | RWTH AACHEN UNIVERSITY

VILLASweb
Overview (cont'd)

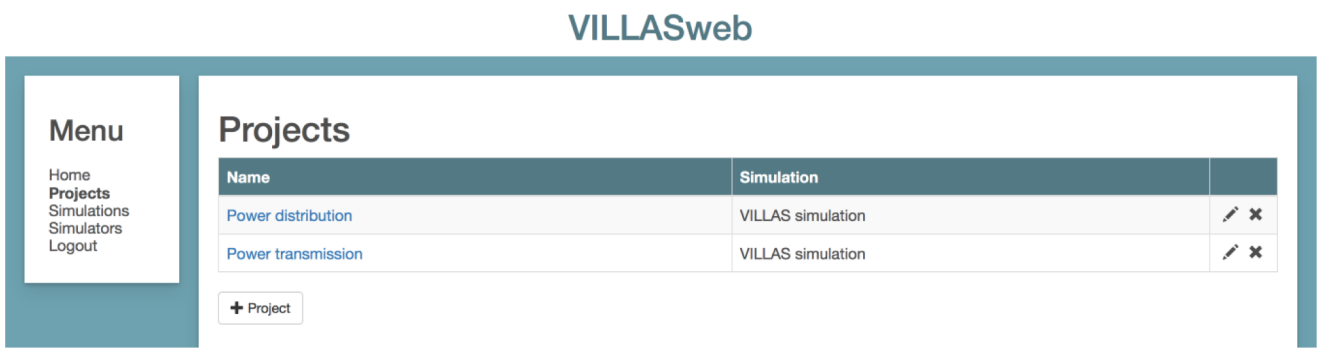
- Menu overview – *Simulations*
  - Multiple *Simulations* can be created and configured for available simulators
  - A *Simulation* refers to a setup of a co-simulation experiment with defined subsystem models
    - For instance, we can create multiple *Simulations* for RT-Super Lab Demo that refer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

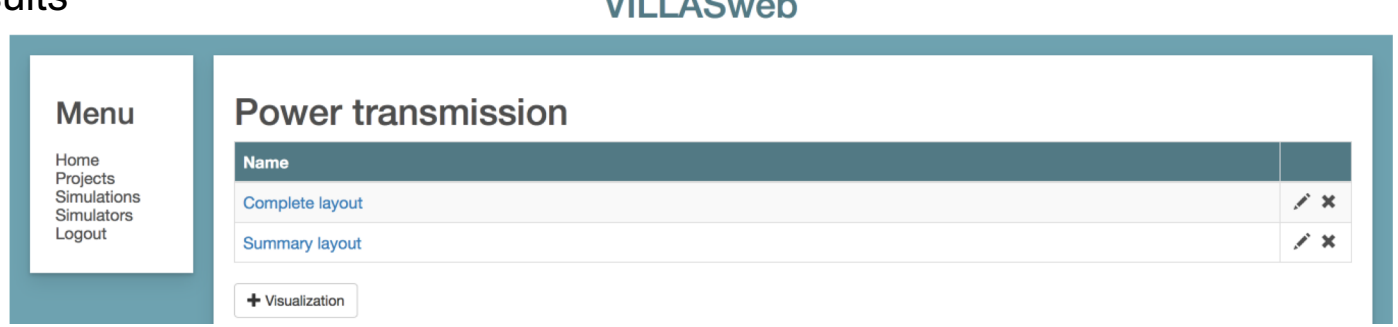


VILLASweb
Overview (cont'd)

- 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

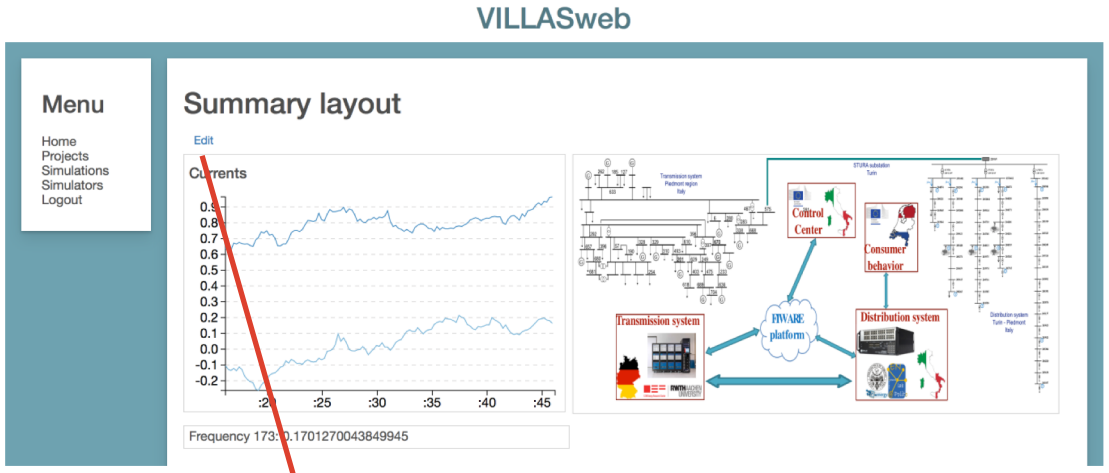


- Menu overview – *Visualizations*
  - Multiple *Visualizations* can be created and configured for a *Project*
    - A user can create different layouts (e.g. Summary or Detailed) for monitoring of simulation results

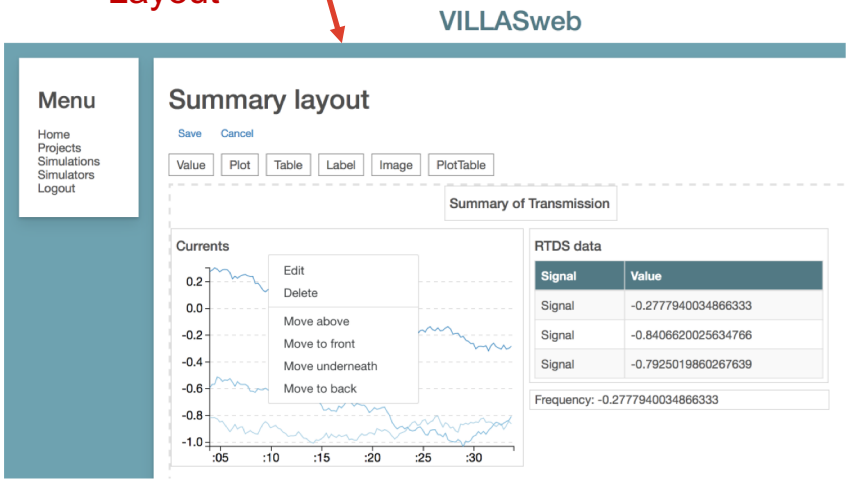


VILLASweb
Overview (cont'd)

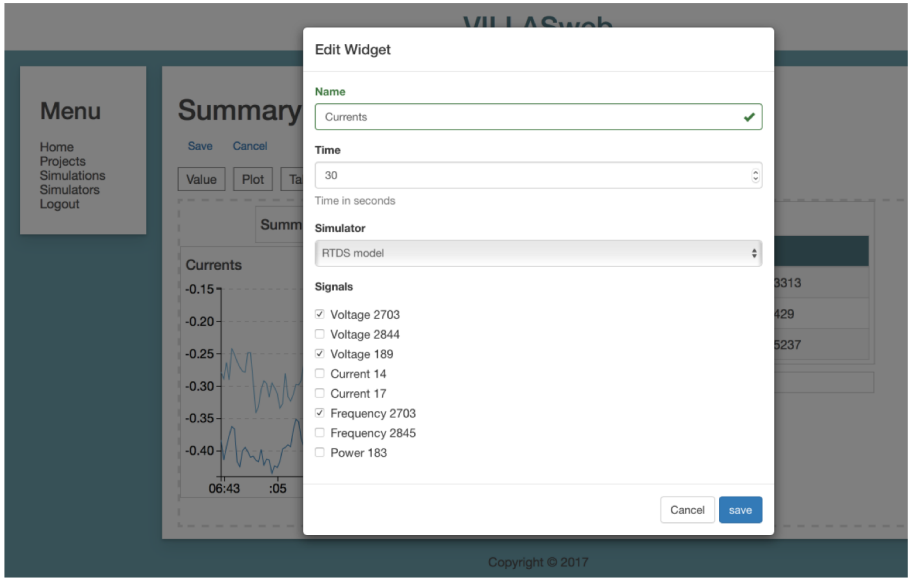
- A user can create a customized visualization of the co-simulation experiment



Edit Visualization Layout



Edit Plot Widget



E.ON Energy Research Center | RWTH AACHEN UNIVERSITY

VILLAScontroller

A JSON-RPC API for controlling distributed simulations

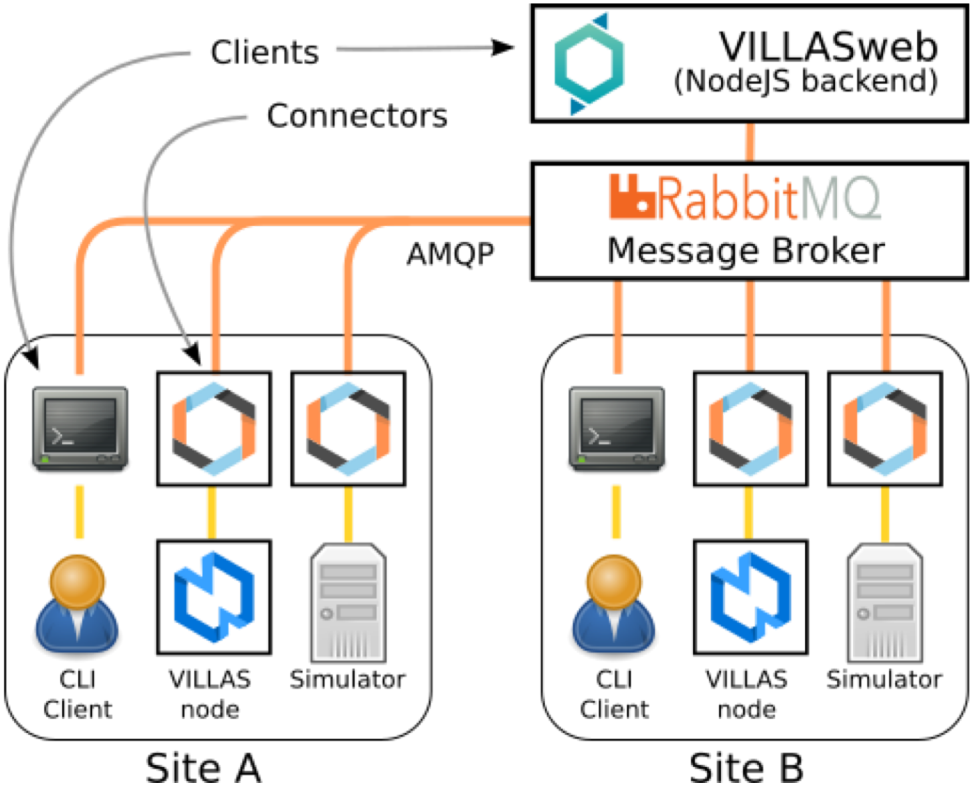
VILLAScontroller

- 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
  - Command Line Tools

- Collection of Tool adapters
  - Mostly implemented as Python scripts

VILLAScontroller Architecture



VILLASnode

A gateway for co-simulation interface data

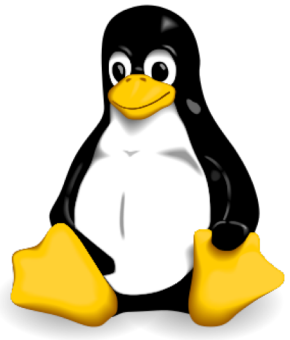
VILLASnode
Basics

- 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/C++ library & set of command line applications
    - You can write your own and just use a subset of the functionality
- Running on Real-time Linux
  - Command line applications
  - Configuration via JSON text file

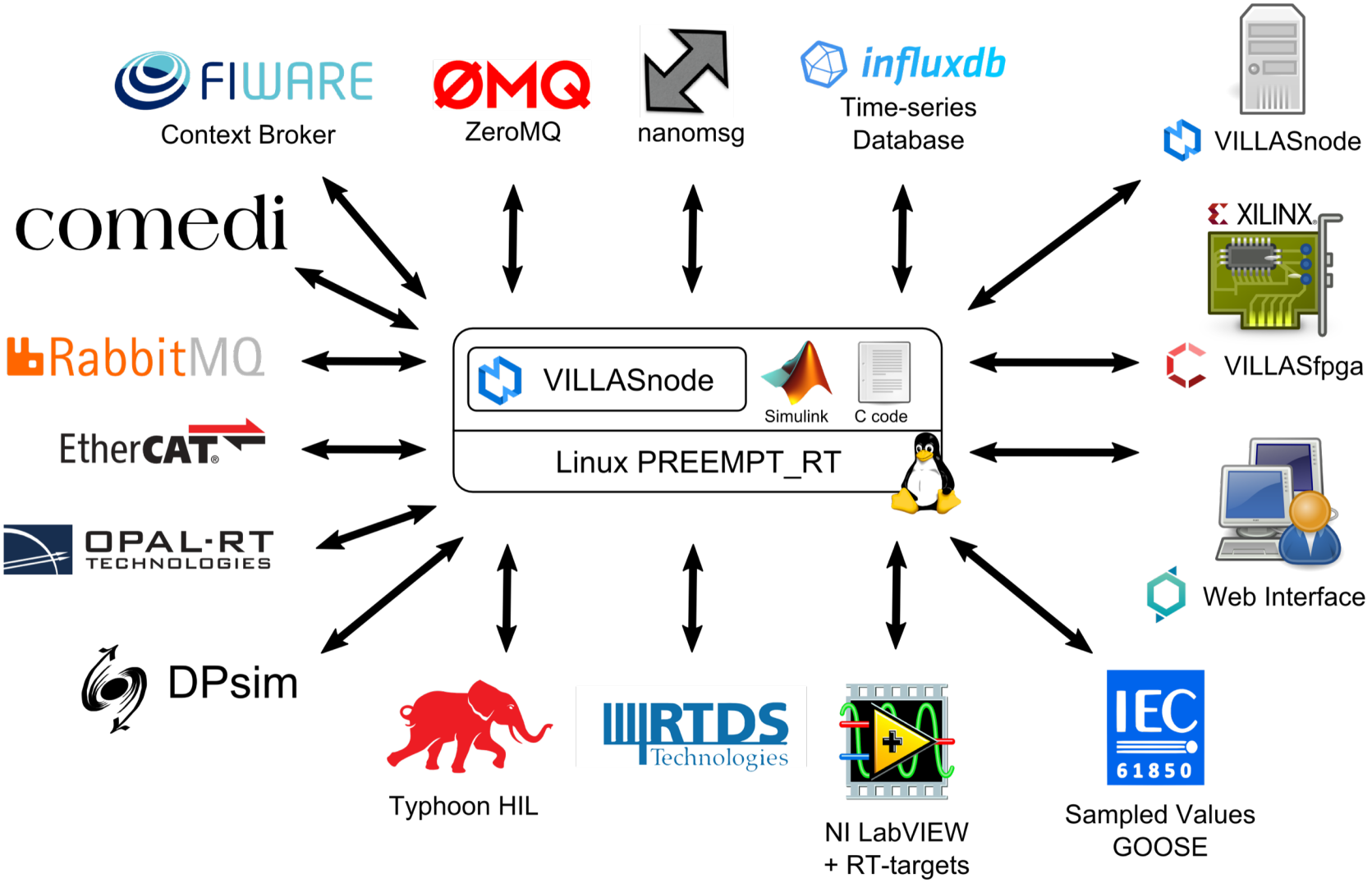
VILLASnode
Design decisions

- 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

- 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

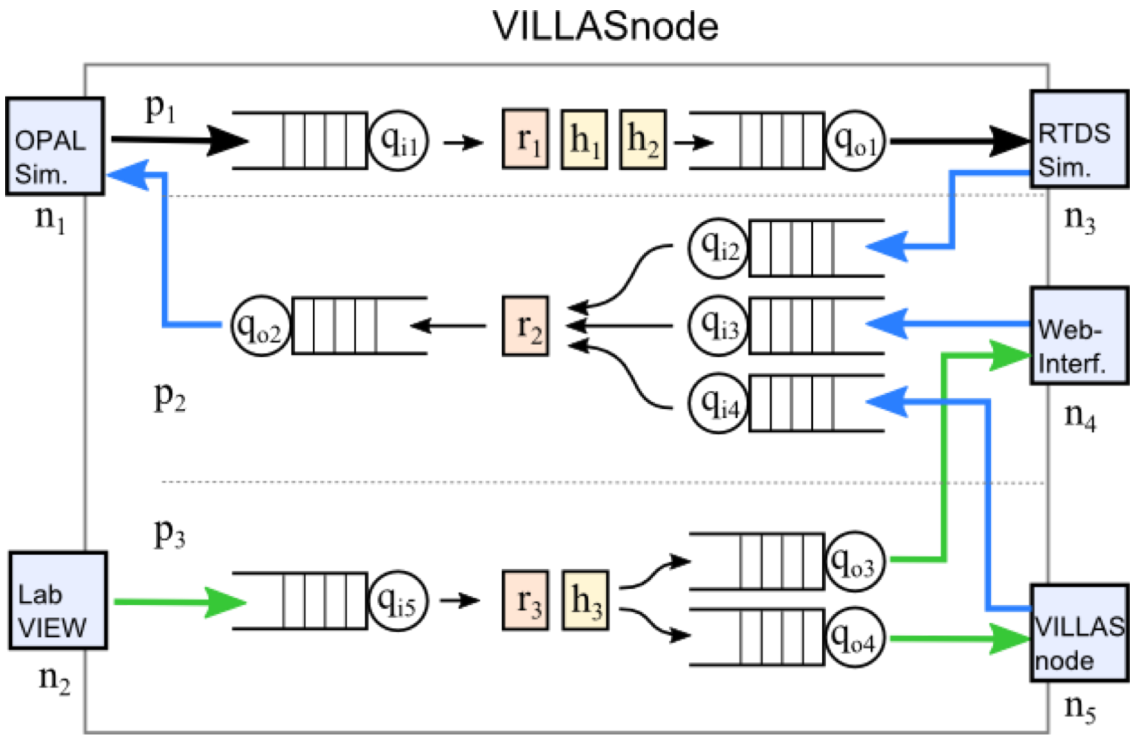


VILLASnode Interfaces



VILLASnode
Concept

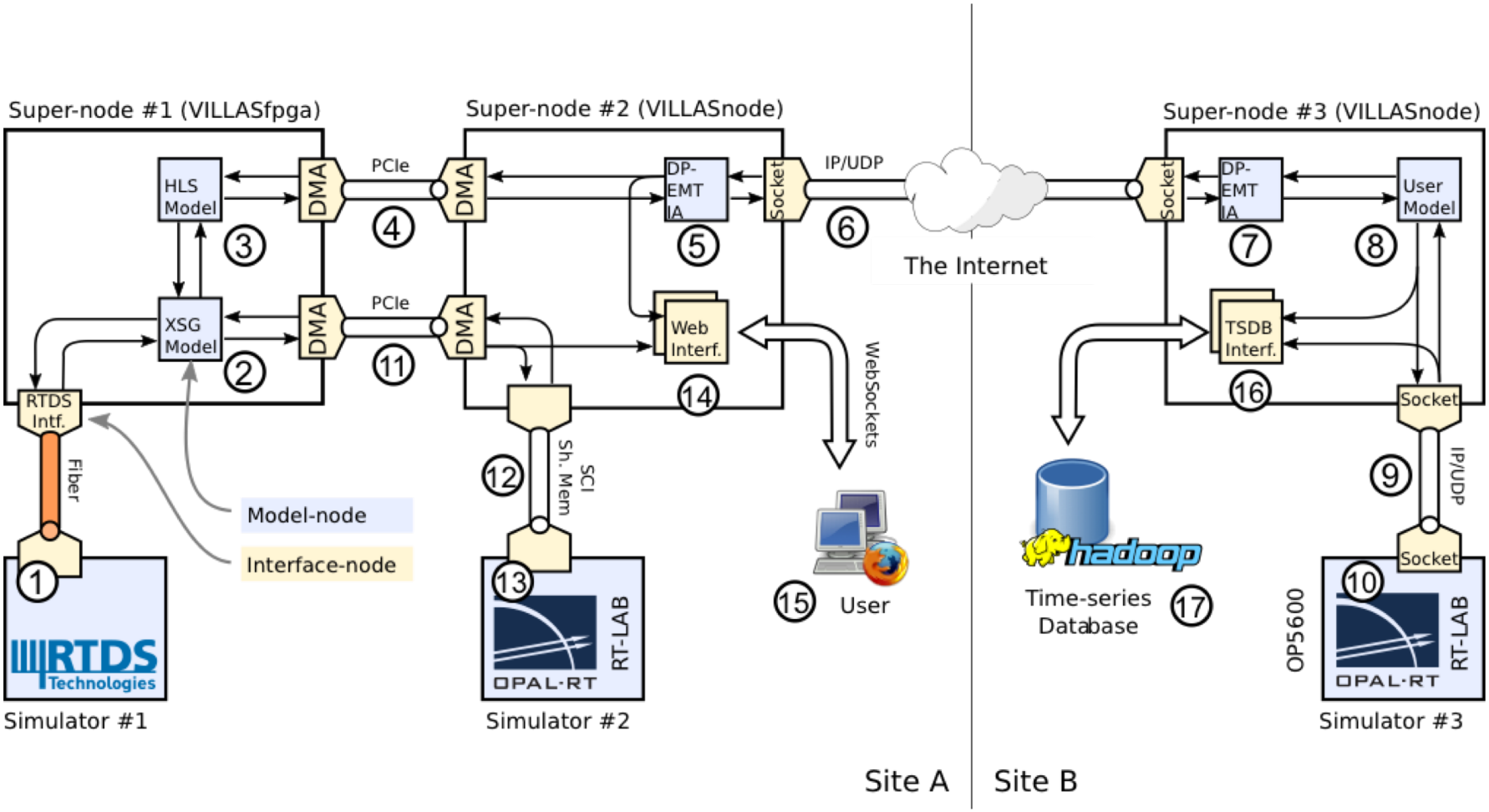
- Router for sample / value based simulation data
  - **n-to-n** forwarding of sample values
  - MUX and DEMUX supported
  - Concept of unidirectional paths (p_i), nodes (n_i), hooks (h_i) & registers (r_i)



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
Flexible topologies



VILLASnode
Features Overview

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

VILLASnode
Tools

- Standard IO streams: stdin / stdout

- Main gateway daemon: `villas node`
  - Executed 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
- Procecesses data from stdin/stdout `villas hook`
- Converts sample data between supported formats `villas convert`

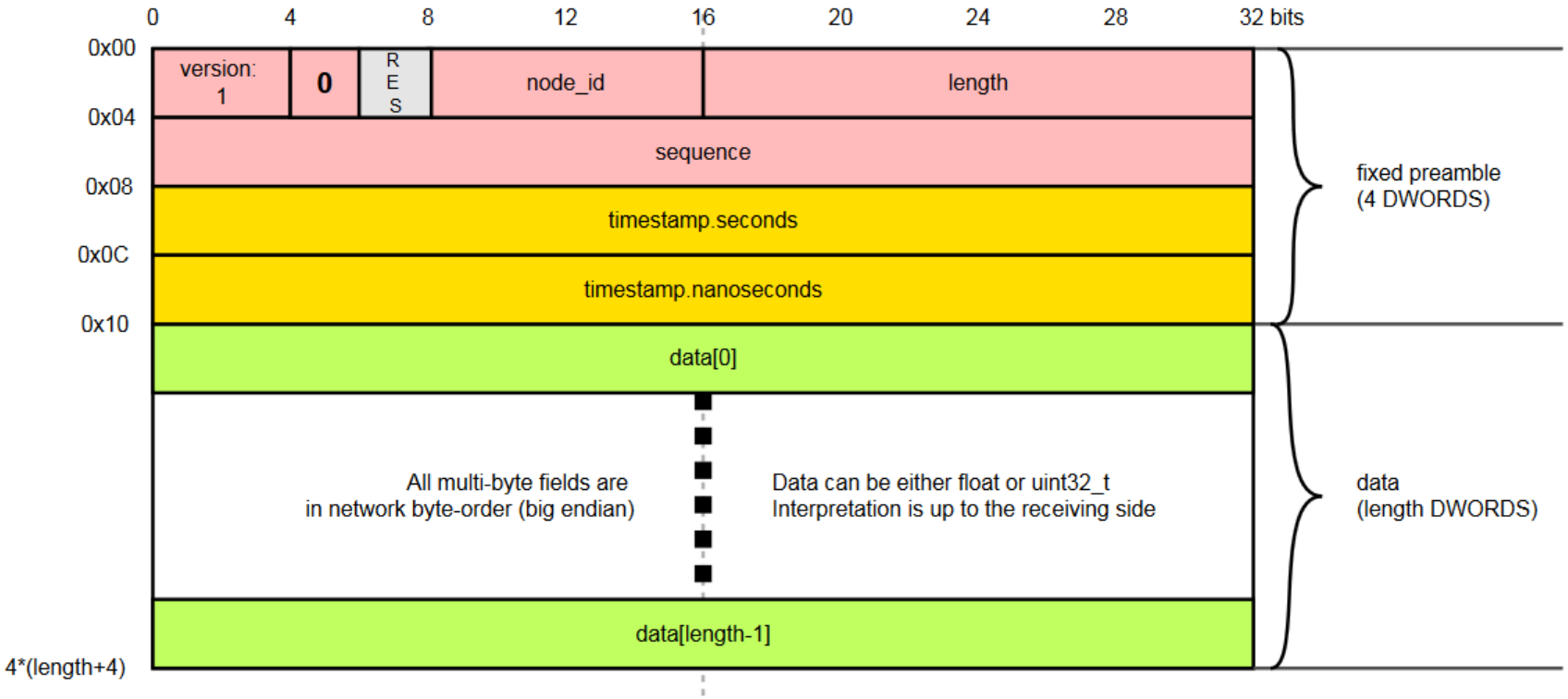
VILLASnode
Interfaces and Protocols

- Separation of Protocols and Payload Formats
- Protocols
  - Standard BSD Sockets: UDP, IP
  - Standard IO: Files
  - AMQP + MQTT
  - ZeroMQ, nanomsg
  - FIWARE NGSI
  - IEC 61850
  - InfluxDB
- Payload Formats
  - RAW Data
  - Protobuf
  - JSON / BSON
  - CSV
  - HDF5?
- Interface = Protocol (+ Payload Format)

```
{
  "ts": {
    "origin": [
      1536647472,
      362173000
    ],
    "received": [
      1536647472,
      362246000
    ]
  },
  "sequence": 44,
  "data": [
    0.022245,
    0.597749,
    true,
    44,
    {
        "real": 23234.2232,
        "imag" : 1200.232
    }
  ]
}
```

VILLASnode
RAW Payload encoding

- Inspired by IP/UDP
- Lightweight encoding
- No metadata



VILLASnode
Google Protobuf

- Space efficient binary encoding
- Like ASN.1 BER → see IEC 61850-8-1

```
syntax = "proto2";

package villas.node;

message Message {
        repeated Sample samples = 1;
}

message Sample {
        enum Type {
                DATA = 1;
                START = 2;
                STOP = 3;
        };

        required Type type = 1 [default = DATA];
        optional uint64 sequence = 2;
        optional Timestamp timestamp = 4;
        repeated Value values = 5;
}
```

```
message Timestamp {
        required uint32 sec = 1;
        required uint32 nsec = 2;
}

message Value {
        oneof value {
                double f = 1;
                int64 i = 2;
                bool b = 3;
                Complex z = 4;
        }
}

message Complex {
        required float real = 1;
        required float imag = 2;
}
```

Synchronization?

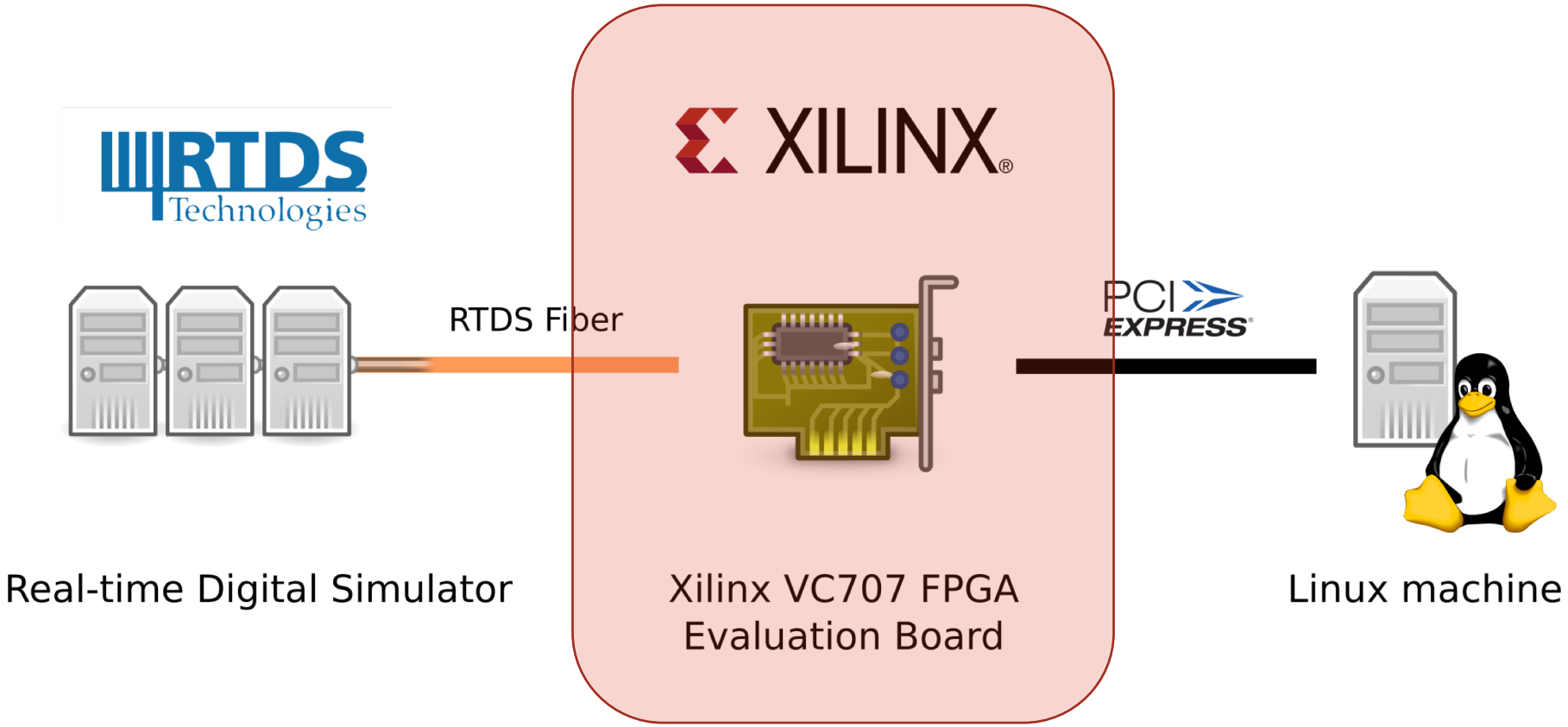
VILLASfpga

A hard real-time interconnect for HiL test benches and local co-simulations

VILLASfpga

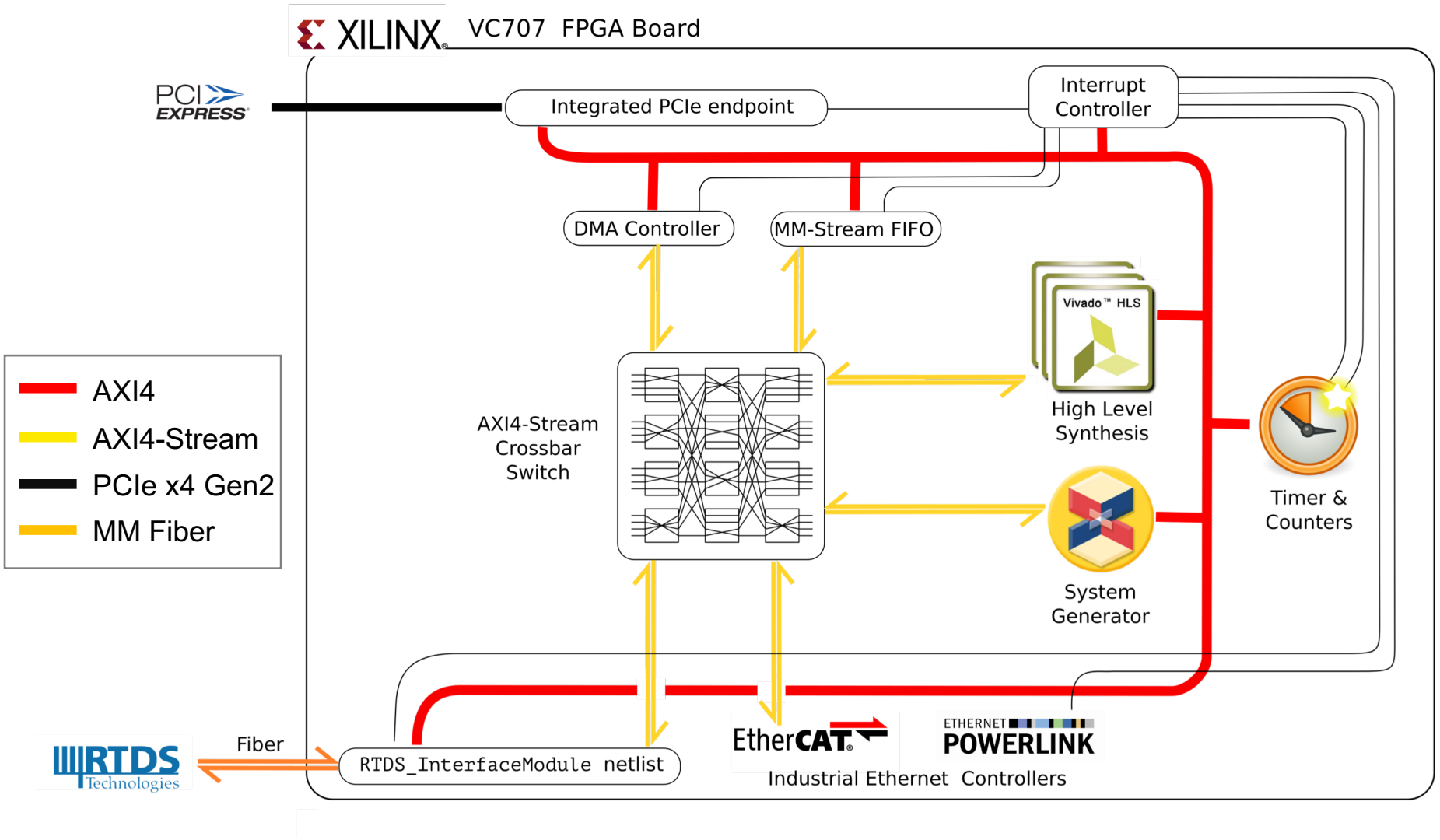
- Extend VILLASnode instances with FPGA ressources
  - Interface DRTS
    - RTDS via GTFPGA netlist
    - Typhoon, OPAL-RT via Aurora protocol
  - Run models / interface-algorithms
    - Simulink / Xilinx System Generator models
    - C++ code using High Level Synthesis

VILLASfpga
Motivation



High-level Architecture

VILLASfpga Architecture



Visibility: FEIN e.V

- Most tools are open source
  - Code on RWTH GitLab
  - Project descriptions on FEIN e.V. website

Förderer der Energie- und Informationstechnik für zukünftige Netze Aachen e.V.

- Already used by external partners:
  - RTE, INL, NREL, SINTEF, POLITO, …

GPL Free Software

- https://www.fein-aachen.org/projects/

VILLASframework

Toolset for distributed real-time simulation and HIL testbed interconnection

Gitlab

DPsim

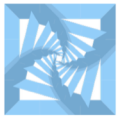
Dynamic phasor power system simulation library in C++

Gitlab

ModPowerSystems

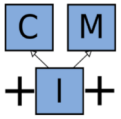
Power systems Modelica library

Gitlab



Pintura

Graphical CIM XML-RDF editor based on new HTML5 / JS / SVG web technologies



CIM++

Deserialiser library for C++ objects from XML/RDF documents based on CIM standards

Github

HermitCore

A Unikernel for Extreme-Scale Computing

Github

Future Goals

- VILLASnode
  - IEC 61850-8-1 (MMS and GOOSE)
  - Distributed Co-simulation Protocol (DCP, ACOSAR project)
  - Functional Mockup Interface (FMI)
  - Real-time Protocol (RTP)
- VILLASweb
  - Tighter integration of Pintura Editor
- VILLAScontroller
  - New Adapters for RTDS and OPAL-RT
- Data Models
  - for Research Infranstruture
  - for Lab Interfaces
  - for Models
- CIM++
  - More CIM converters (RTDS, Simulink)
- DPsim
  - Converter Models
  - Dynamic Phasor Simulation with multiple harmonics

Contributions welcome!

Demo Time

A Demo Scenario

- Western System Coordinating Council Benchmark Grid
  - 9 Busses
  - 3 Generators
- Dynamic Model
  - Classic Transient Stability Model

- Scenario
  - Fault at BUS9
  - Changed inertia of GEN3 (13,8 kV)
  - Load change

- A full workflow:
  - Model entry/edit in Pintura
  - Simulation planning and execution in VILLASweb
  - Analysis in Jupyter

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