

Data Modeling for Hardware Component Information in Testbeds

Motivation

At the Chair of Network Architectures and Services, we host various testbeds for network experiments. We employ a self-developed testbed methodology called "pos" to conduct our experiments. This methodology documents all relevant data and metadata associated with each experiment. Alongside experimental data, recording the hardware components of the experiment hosts is crucial for comprehensive documentation. This hardware information simplifies the reproduction of experiments and ensures consistent results.

Currently, we use a proprietary JSON-based format to describe hardware information. As part of our commitment to open science, we aim to explore how YANG–known for its ability to offer a standardized and machine-readable representation of data–can enhance the management and understanding of the hardware used in our experiments. Yet Another Next Generation (YANG), standardized in RFC 6020, is a data modeling language designed for the NETCONF and RESTCONF protocols. In recent years, many data models for various use cases have been proposed.

Hostname VM Access Group	Processor	Cores	Memory	Network Interface
// expr.	// expr.	# int	GB int	// expr.
ada (fab resting)	AMD EPYC 7542 AMD EPYC 7542	64	1024	Intel 82599ES 10-Gigabit SF/ISFP+ englised0 englised1 Intel 8810-C for QSFP englised0 Intel E810-C for QSFP englised0 Intel E810-XW for SFP englised0 Intel S00-XW for SFP englised0 Intel X0-Y0 for ZSCES Intel X0-Y0 for ZSCES
algofi algorand slicestestbed	AMD EPYC 7543	32	512	Broadm BCM57416 NetXtreme-E Dual-Media 10G RDMA (enoting) (enoting) Intel 82580 Gigabit (enp131600) (enp131601) (enp131601) Intel E810-Xf Vor QSEP (enp138600) (enp138601) Intel E810-XXV for SEP (enp138600) (enp138601)

Visualization of hardware information on the testbed's website

Your Task

- Familiarize yourself with YANG (RFC 6020) and the IETF Hardware Model for YANG (RFC 8348)
- Propose/Extend a data model for testbed-specific hardware information
- Implement the data model in our testbed infrastructure
- Demonstrate the functionality in a prototype

Eric Hauser hauser@net.in.tum.de



Contact

