

# Tomato

tomato.german-lab.de

a network experimentation tool

Dennis Schwerdel  
University of Kaiserslautern, ICSY Lab

SPONSORED BY THE



Federal Ministry  
of Education  
and Research

# Overview

- ▶ Introduction
- ▶ ToMaTo
  - Architecture
  - Devices
  - Connectors
  - Features
  - Editor
- ▶ Evaluation

The screenshot displays the G-Lab ToMaTo web interface in a browser window. The browser address bar shows 'localhost:8080/top/2/edit'. The page header includes the G-Lab ToMaTo logo and navigation links for 'Topologies', 'Admin', 'Help', and 'Create Ticket'. The main content area is titled 'Edit topology 2' and features a sidebar with various topology management tools: Wizard, OpenVZ, KVM, External, Hub, Switch, Router, Layout, and Remove. The main workspace shows a network diagram with a central 'switch1' connected to a 'server' above it, and two 'client' nodes ('client1' and 'client2') connected to the switch. The footer contains the G-Lab logo and copyright information: 'Copyright © 2010 University of Kaiserslautern, on behalf of the German Lab, see [license page](#) for details.'

# German-Lab Project

---

- ▶ National future-internet research project
  - Funded by BMBF
  - Phase 1: One big project, 6 partner universities
  - Phase 2: Multiple small projects, some more partners (also industry)
  
- ▶ Integrated with an experimental facility
  - Facility users and providers are the same people
  - Experimental facility providers form special interest group
  - 180+ nodes in total
    - Mostly at 6 phase 1 sites
    - Some special equipment from phase 2
  - No layer-2 connectivity
  
- ▶ Currently closed platform
  - MoUs
  - Federation plans (esp. GENI)
  - Commercialization planned after end of phase 1

# Introduction

---

## ▶ Why experimental facilities?

- Helps designing, testing and improving network algorithms, applications and architectures
- Instrument for networking research

## ▶ Existing experimental facilities

- Planet-Lab
  - High node numbers → allows large experiments
  - Uses container virtualization → limited hardware and kernel access
  - Shared network interfaces → no network topologies
- Emulab
  - No virtualization → full hardware and kernel access, inefficient
  - Realized as cluster → full control over network topology
- Seattle
  - Very high node numbers → allows large experiments
  - Uses virtual python interpreter → no operating system access
  - Only custom python dialect → not suitable for all experiments

# ToMaTo

▶ ToMaTo: „Topology Management Tool“

▶ Topology contains

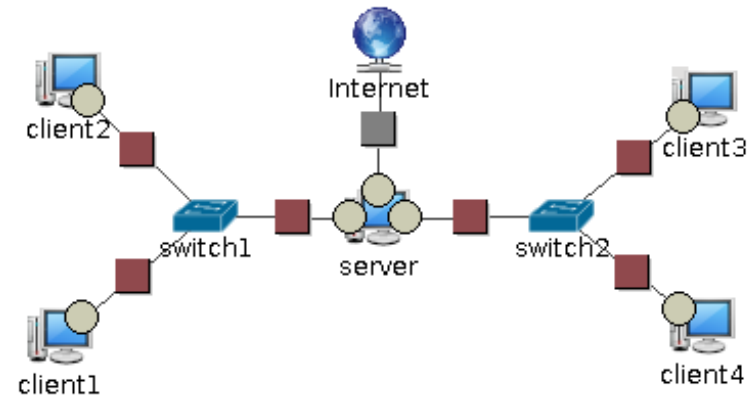
- Devices
- Connectors

▶ Devices

- Active components
- E.g. computers
- Produce/Consume data

▶ Connectors

- Networking components
- E.g. switches, routers
- Transport/Manipulate data

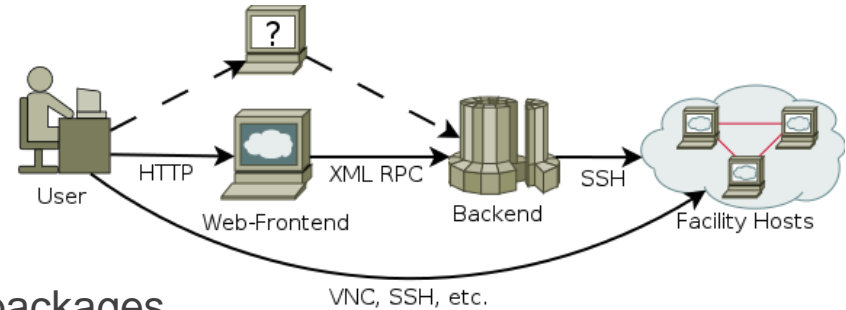


# ToMaTo - Architecture

- ▶ ToMaTo contains 3 parts

- ▶ Host part

- Based on PROXMOX VE
- Offers virtualization
- All additional software available as packages



- ▶ Backend

- Controls hosts via SSH
- Centralized logic, resource management, user accounts
- Offers XML RPC interface

- ▶ Frontend(s)

- Offer a GUI to users
- Currently only a web-based interface exists
- More frontends possible

# ToMaTo - Devices

---

- ▶ KVM
  - Full virtualization
  - Offers full access
  
- ▶ OpenVZ
  - Container virtualization
  - Lightweight, uses less resources
  - Easier to access
  
- ▶ Templates
  - Preinstalled device images, ready to use
  - Current versions of Debian and Ubuntu Linux (32 and 64 bits)
  - FreeBSD (only KVM)

# ToMaTo - Connectors

---

- ▶ Tinc VPN
  - Creates private networks between devices
  - Not encrypted → good performance
  - Hub, Switch or Router semantics
  
- ▶ External networks
  - Connect devices to host bridges
  - Can connect devices to the Internet
  - Openflow integration
  
- ▶ Link emulation
  - Only available for Tinc VPN
  - Can set additional QoS limitations on links
    - Packet loss
    - Bandwidth limit
    - Latency
  - Long-term statistics help users to estimate underlying link properties



# ToMaTo - Features

---

- ▶ Administrator/Developer features
  - Intelligent load-balancing
  - Open xml-rpc interface
  - Administrator tools
  - LDAP integration
  
- ▶ User features
  - Automatic network interface configuration
  - Changes to running topologies
  - Console access
  - Image up/download
  - Pcap capturing

# Evaluation

---

- ▶ Methodology
  - Identify network experiment classes
  - Evaluate ToMaTo support for these classes
- ▶ Access layer experiments
- ▶ Network layer experiments
- ▶ Algorithm/Protocol experiments
- ▶ Legacy software experiments

# Evaluation - Access layer experiments

---

- ▶ Topic: lower layers and hardware
  - Example: Mobile handover
  
- ▶ Requirements
  - Hardware access
  - Custom operating systems (Realtime)
  - Heterogeneous access technologies (3G, Wifi, etc.)
  - Complex setup, depending on hardware type
  
- ▶ Needs specialized testbeds depending on hardware
  - DES Testbed, Wisebed
  
- ▶ Not supported in ToMaTo

# Evaluation - Network layer experiments

---

- ▶ Topic: TCP/IP suite
  - Example: IPv6 extensions, TCP substitutes
- ▶ Requirements
  - Deep OS access (modified kernels, etc.)
  - Small but complex topologies, link emulation
- ▶ ToMaTo offers
  - Full kernel access via KVM
  - Complex topologies
  - Link emulation
  - Packet capturing (for analysis)
  - Easy setup of topologies

# Evaluation - Algorithm/Protocol experiments

---

- ▶ Work on top of network layer
  - Example: P2P-Networks
  
- ▶ Requirements
  - Huge but simple topologies
  - Link emulation
  - No hardware or OS access
  
- ▶ ToMaTo offers
  - Lightweight virtualization with OpenVZ (250 devices per host)
  - Link emulation
  - Ability to connect to other testbeds via Internet
  - Up/download of images (simplifies setup of multiple devices)

# Evaluation - Legacy software experiments

---

- ▶ Topic: legacy software
  - „Legacy software“ refers to any widespread software with undocumented or unpublished behavior
  - Example: Skype and Windows
  
- ▶ Requirements
  - Special environments, custom operating systems
  - Access to external services
  - Small but complex topologies
  - Link emulation and external packet capturing
  
- ▶ ToMaTo offers
  - Custom operating systems with KVM (even Windows and BSD run)
  - Access to external service via Internet connector
  - Packet capturing independent of guest OS

# ToMaTo Summary

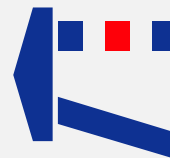
---

- ▶ Flexible and extensible
  - 3-tier architecture
  - Openflow integration
- ▶ Efficient
  - As much access as needed (KVM)
  - As less resource usage as possible (OpenVZ)
- ▶ Easy to use
  - Graphical editor, topology creator
  - Automatic interface configuration
- ▶ Supports experiments
  - Ready-to-use templates
  - Link emulation
  - Image download/upload
  - Packet capturing
  - Direct console access

# Thank you



TECHNISCHE UNIVERSITÄT  
KAISERSLAUTERN



**Dipl.-Inf. Dennis Schwerdel**

**Integrated Communication Systems ICSY**

**University of Kaiserslautern**

**Department of Computer Science**

**P.O. Box 3049**

**D-67653 Kaiserslautern**

Phone: +49 (0)631 205-26 43

Fax: +49 (0)631 205-30 56

Email: [schwerdel@informatik.uni-kl.de](mailto:schwerdel@informatik.uni-kl.de)

Internet: <http://www.icsy.de>