MINER

Modular Infrastructure for Networked Experimentation and Research

Motivation

Distributed performance evaluation and testing typically require the orchestrated execution of multiple activities in remote locations. In a systematic study a significant number of different configurations has to be investigated and all experiments need to be documented persistently. The ability the repeat the complete set of experiments at virtually no cost is essential. The 'manual' execution without any form of a supporting control system is a tedious and error prone process that clearly fails to scale.

Overview

The main objective of MINER is to support users in conducting distributed experiments. To this end, MINER is a programmable platform that significantly simplifies the orchestrated usage of arbitrary tools via a unified application programming interface (API).

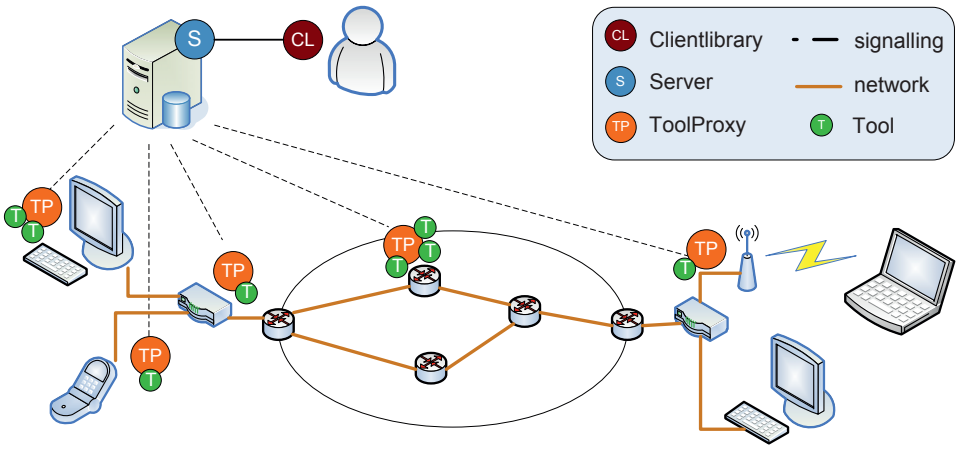
The infrastructure enables a user to

- specify experiments, so called MINER scenarios
- schedule the execution of scenarios
- retrieve results and associated logs of scenario executions

MINER does not reinvent the wheel by re-implementing existing tool functionality. Instead, a core concept of MINER is the integration of existing tools. MINER thus provides a tool integration API that enables the development of a MINER tool which wraps an existing tool executable. The MINER tool can then be plugged into the MINER platform.
A user can conveniently access the MINER functionality by a client library. It is tool-agnostic in the sense that – independent of the native tools' characteristics – there is a unified API that is used to employ tools in an experiment, configure them and retrieve their results.

Main features

- Programmable platform for distributed experimentation
- Convenient access through a tool-agnostic client library
- Orchestrated execution of experiments
- Detailed error handling and reporting
- Persistent storage of experiments, executions, and their results
- Allows for complex experiment specifications comprising a multitude of activities
- Time-based and event-based (de-)activation of activities
- Plugin interface for simple integration of existing tools
- Real-time evaluation of conditions and alarm generation



Design

As shown in the figure, the MINER architecture comprises the following components

- 1 Server that is the core of the system
- a number of ToolProxies that are located on the distributed nodes
- MINER tools that are plugged into the ToolProxies
- a client library for convenient programmatic access

A developer makes use of the client library to communicate (exclusively) with the Server which strictly verifies scenario specifications and accepts scheduling requests for scenario executions. When the time is due, the Server triggers and orchestrates the execution. This involves distributing the scenario to the ToolProxies, ensuring the correct initialization of all tools, collecting and saving results, and generally detecting and handling errors properly.

Usage

MINER has so far been successfully deployed in various contexts to implement

- a policy-driven autonomic SLA-validation system (EU project NETQOS)
- multi-layer QoS and QoE monitoring in an Austrian ISP cable network
- regression testing of a newly developed transport protocol
- reliability testing of a proprietary signalling protocol used in a safety-critical environment
- performance evaluation of a distributed synchronization framework
- permanent QoS monitoring of a network used for safety-critical applications
- combined application/network performance evaluation of a logistics software operated world-wide

Availability

MINER is provided under a proprietary license from Salzburg Research. We will happily discuss concrete usage scenarios as well as requests for client specific extensions and consulting services.

DI Christof Brandauer
Salzburg Research Forschungsgesellschaft mbH
Jakob-Haringer-Str. 5/3 | 5020 Salzburg, Austria
T +43.662.2288-447 | F -222
christof.brandauer@salzburgresearch.at
miner.salzburgresearch.at | www.salzburgresearch.at

CONTACT