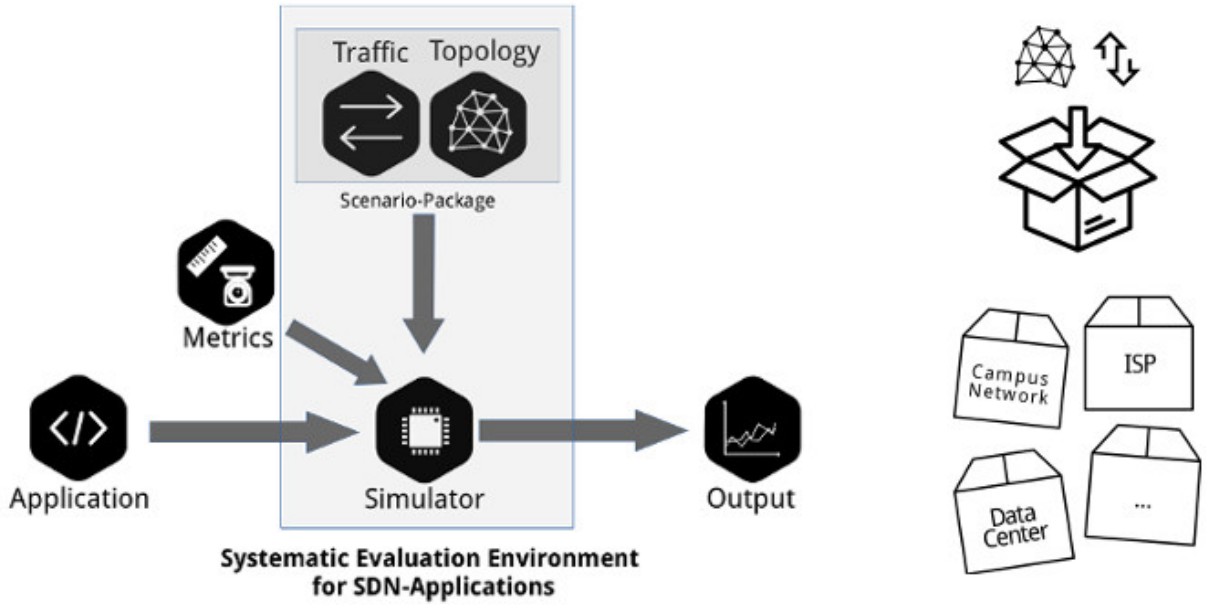
Karlsruhe Institute of Technology

# SEED: Towards a Systematic Evaluation Environment for Software-Defined-Network Applications

9th SDN Switzerland Workshop, Zürich

Addis Dittebrandt, **Michael König**, Felix Neumeister | December 4th, 2017

INSTITUTE OF TELEMATICS — DEPARTMENT OF INFORMATICS — KARLSRUHE INSTITUTE OF TECHNOLOGY



# Praxis der Forschung

"Research in Practice":
Student research project

- Duration: 2 semesters
- Identify state of the art
- Write project application
- Conduct research project
- Write research paper


Source: xkcd.com

# Central Question

Karlsruhe Institute of Technology

**How are SDN-applications evaluated and how can this evaluation process be simplified?**

# Goal

Karlsruhe Institute of Technology

**Focus of the overall project:**

Systematic Evaluation Environment for SDN Applications

**Focus of this presentation:**

Short Project Presentation

+

Call-for-participation

# Outline

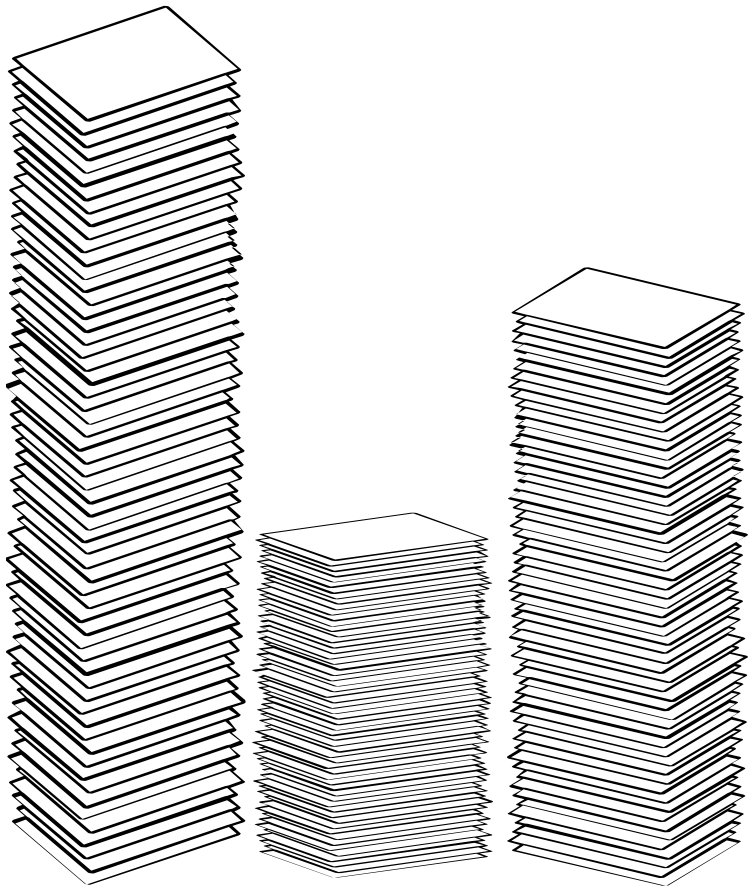
- Motivation: Challenges with SDN-evaluations
  - Reproducibility
  - Comparability
  - Setup-overhead
- Current way to evaluate simulatively
- Our approach: SEED
  - Idea / Architecture
  - Components
  - Workflow
- Conclusion

# Motivation

## Increasing number of SDN-applications & -research

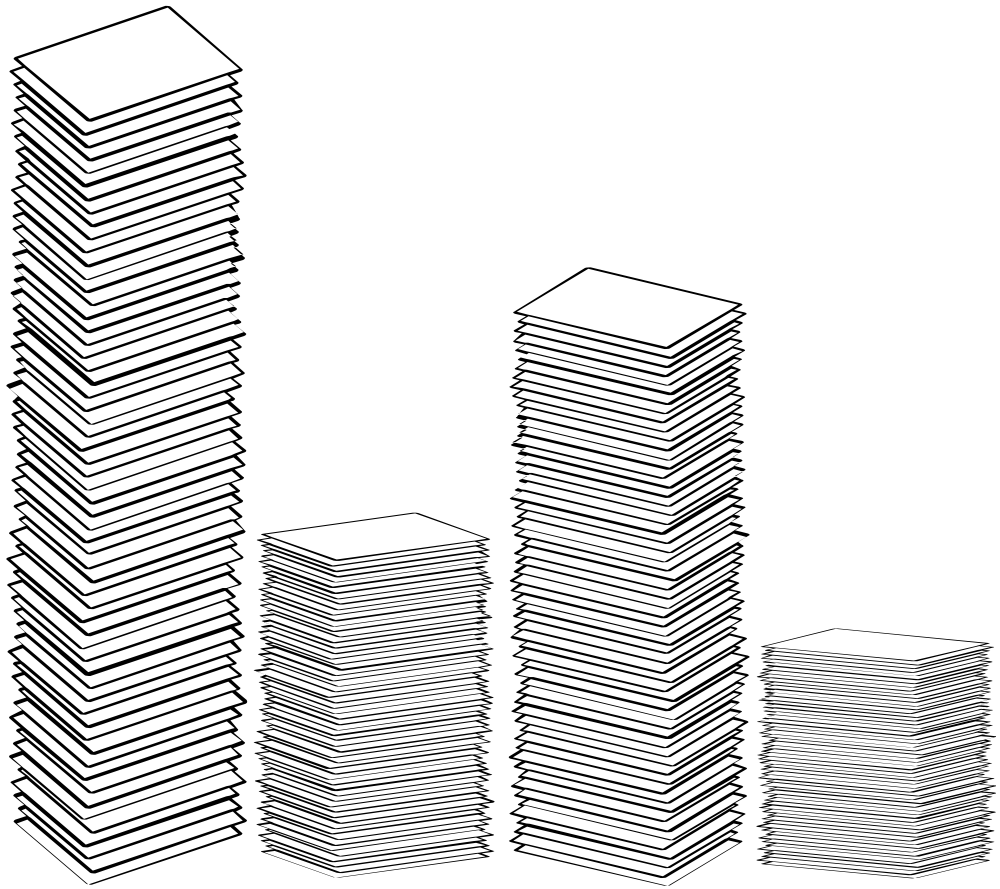
# Motivation

## Increasing number of SDN-applications & -research

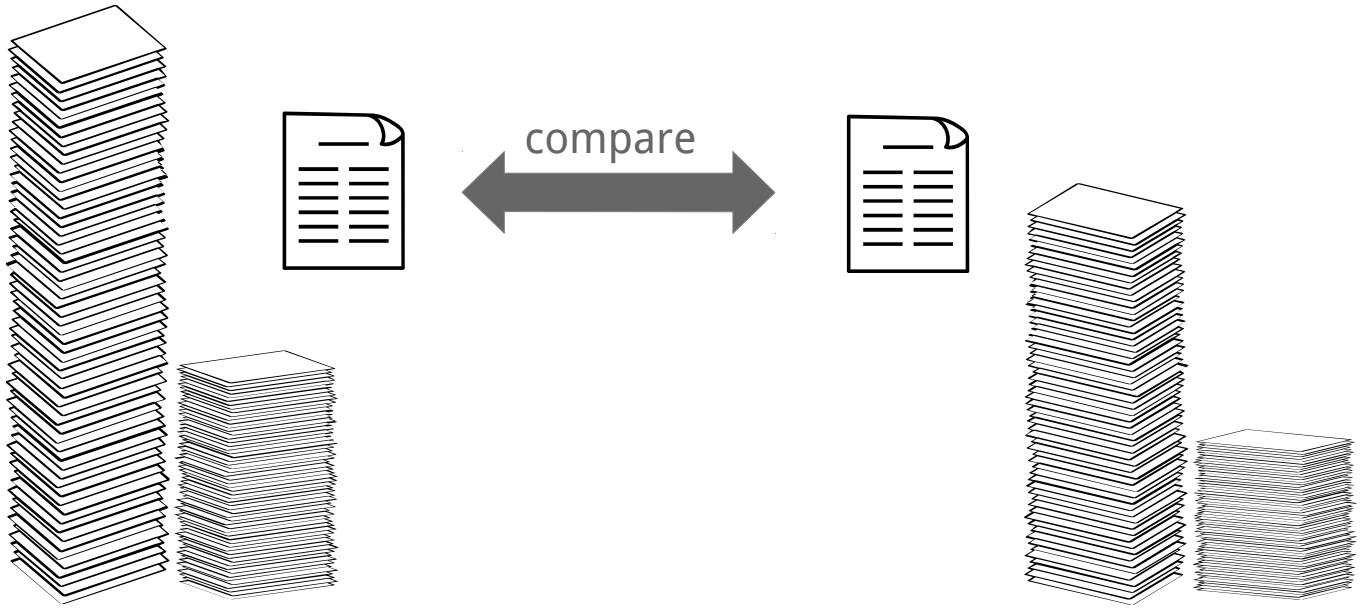


# Motivation

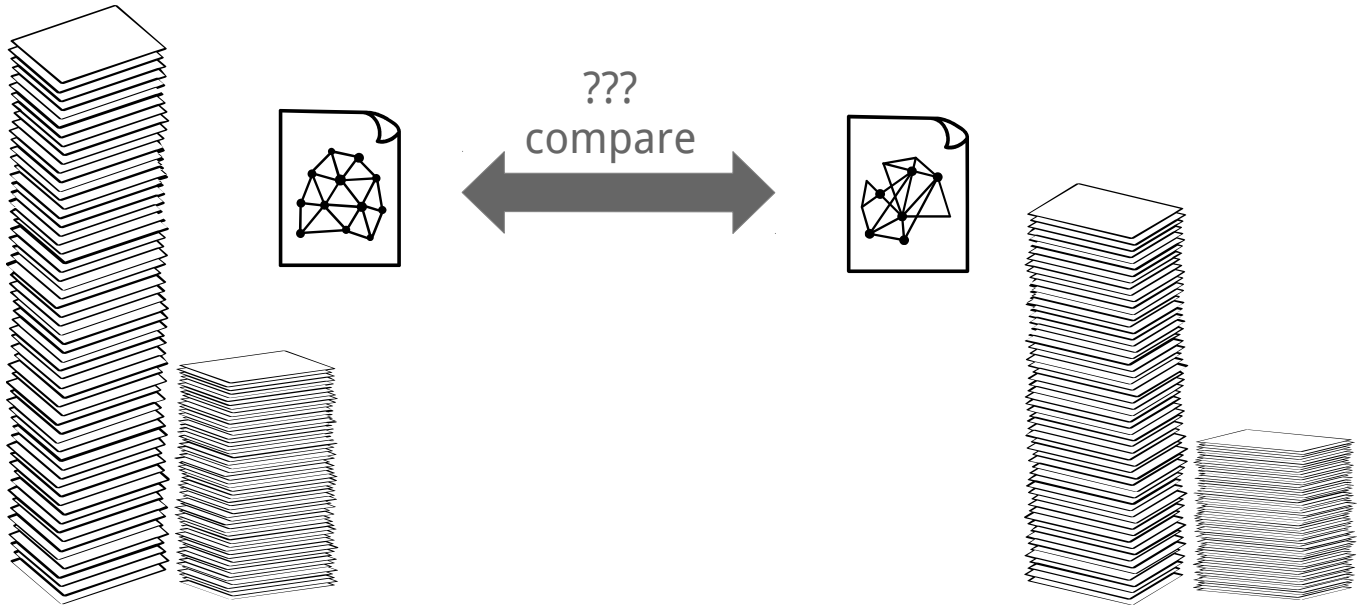
## Increasing number of SDN-applications & -research



# Lack of Comparability



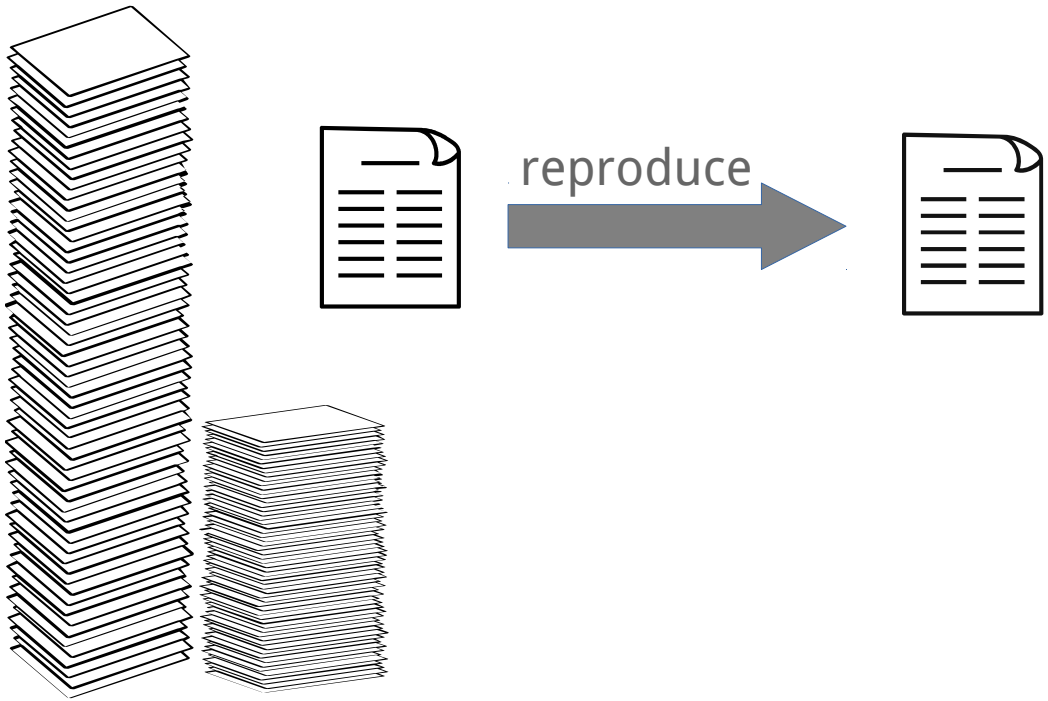
# Lack of Comparability



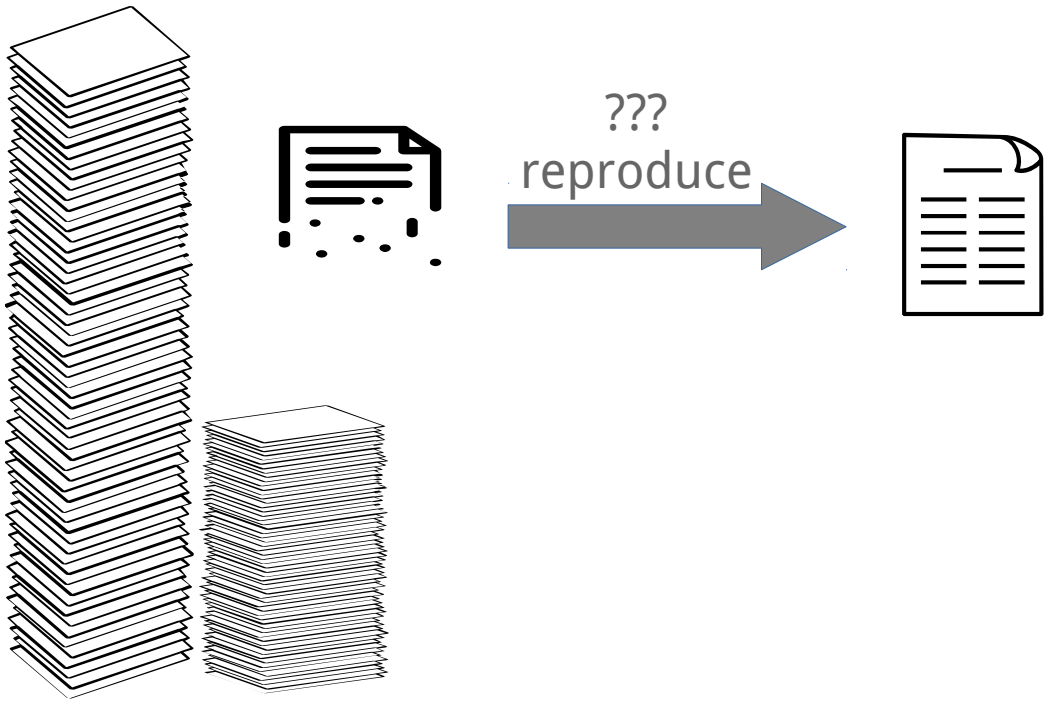

Comparison between SDN-applications often not trivial or infeasible:

- Scenario description unclear or not existing . . .
- Usage of different topologies, traffic, metrics, parameters . . .
- Commonly used scenarios (e.g. "'Stanford Campus Network"') only equivalent in name

# Reproducibility of Results



# Reproducibility of Results




Reproducibility of SDN-applications often not feasible:

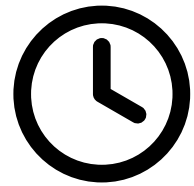
- Contradicting experiment descriptions
- Unclear evaluation setup (missing information)
- Broken, incomplete or missing artifacts

# Evaluation Setup is Complex

Karlsruhe Institute of Technology

- Selection and configuration of tools to generate
  - representative traffic
  - representative topologies

  can be time consuming and error-prone

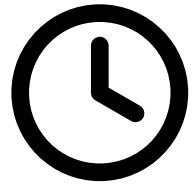


# Evaluation Setup is Complex

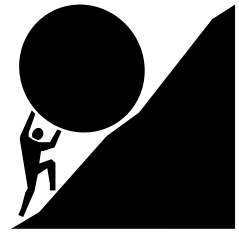
Karlsruhe Institute of Technology

- Selection and configuration of tools to generate
  - representative traffic
  - representative topologies

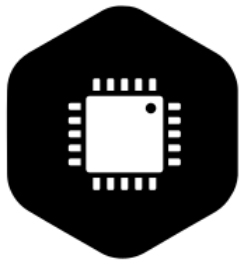
  can be time consuming and error-prone



- Evaluation design has to be re-done
  by each research-team

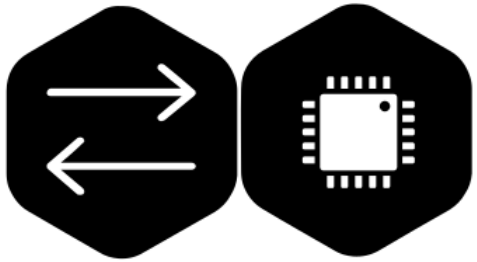


# Current way to evaluate simulatively



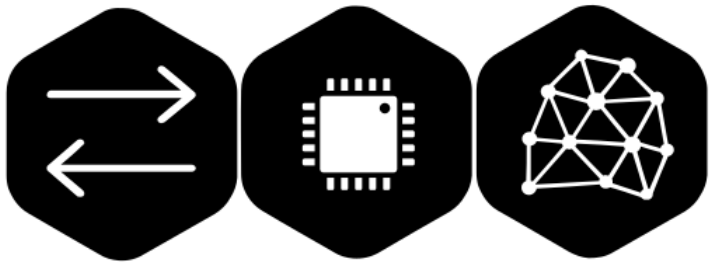
- Simulator
- Traffic
- Topology
- Metrics
- SDN-Application
- Logs/Artifacts

# Current way to evaluate simulatively



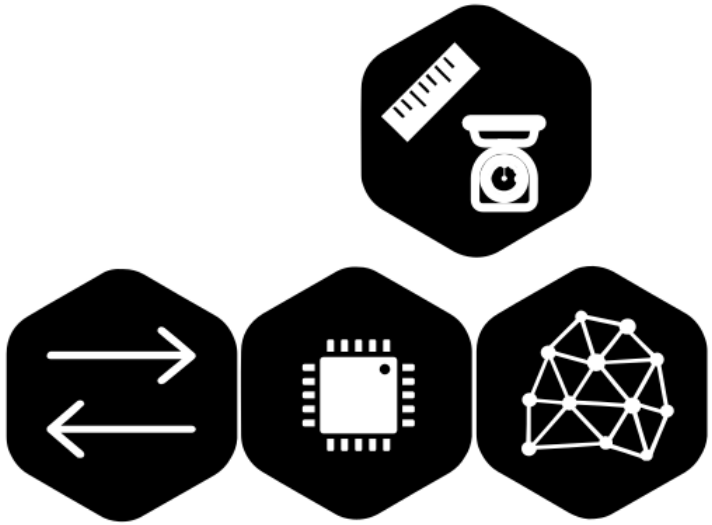
- Simulator
- Traffic
- Topology
- Metrics
- SDN-Application
- Logs/Artifacts

# Current way to evaluate simulatively



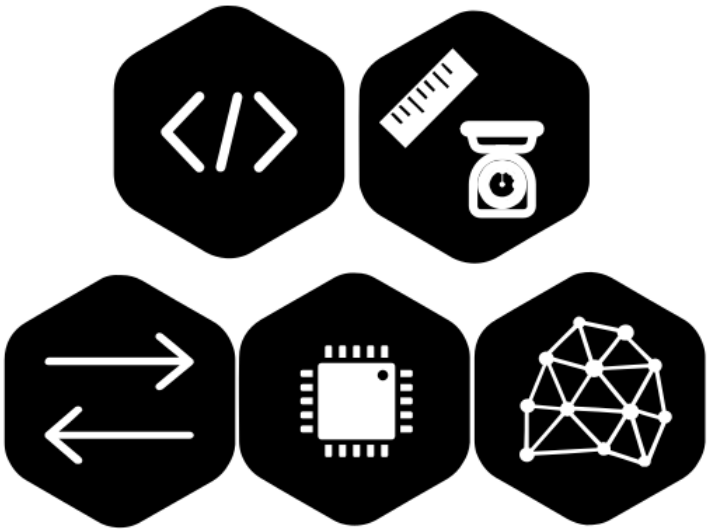
- Simulator
- Traffic
- Topology
- Metrics
- SDN-Application
- Logs/Artifacts

# Current way to evaluate simulatively



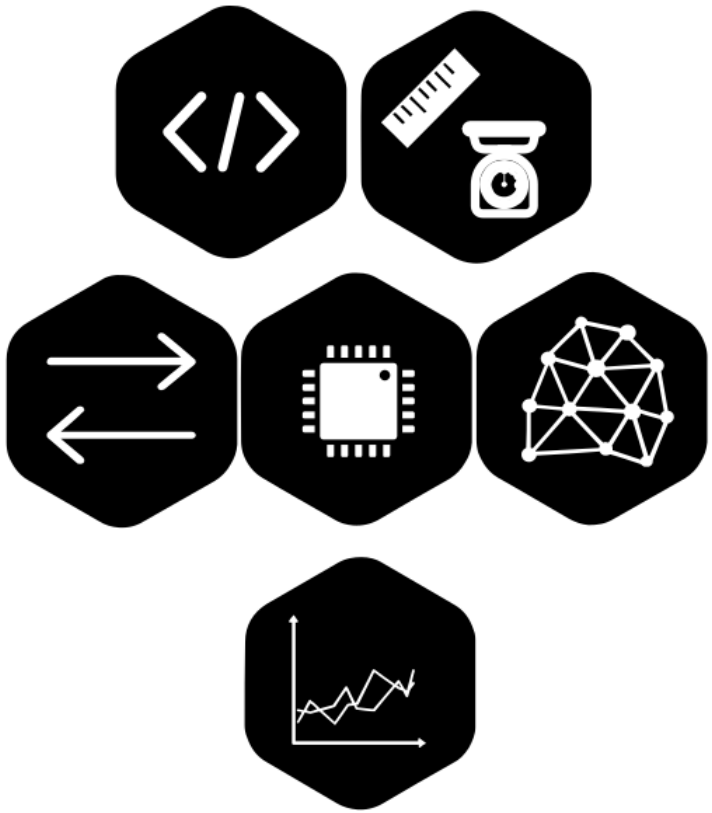
- Simulator
- Traffic
- Topology
- Metrics
- SDN-Application
- Logs/Artifacts

# Current way to evaluate simulatively



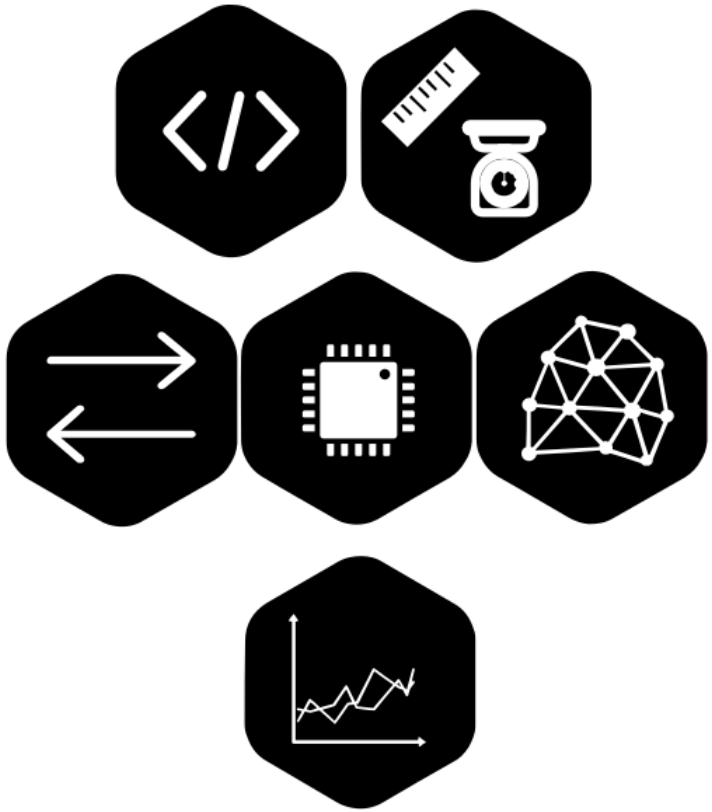
- Simulator
- Traffic
- Topology
- Metrics
- SDN-Application
- Logs/Artifacts

# Current way to evaluate simulatively



- Simulator
- Traffic
- Topology
- Metrics
- SDN-Application
- Logs/Artifacts

# Current way to evaluate simulatively



- Simulator
- Traffic
- Topology
- Metrics
- SDN-Application
- Logs/Artifacts

$\Longrightarrow$ one coupled "blob"

# Challenges of (SDN-)evaluations

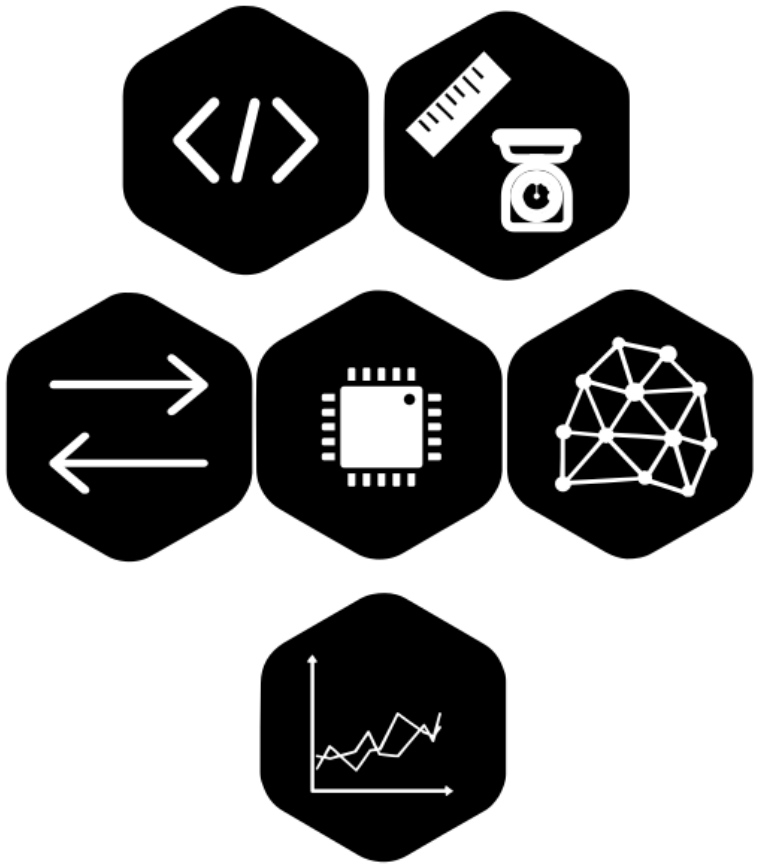
Karlsruhe Institute of Technology

- Lack of Comparability
- Reproducibility often not feasible
- Evaluation-Setup is time consuming & error-prone
  (+ Sharing of existing setups difficult)

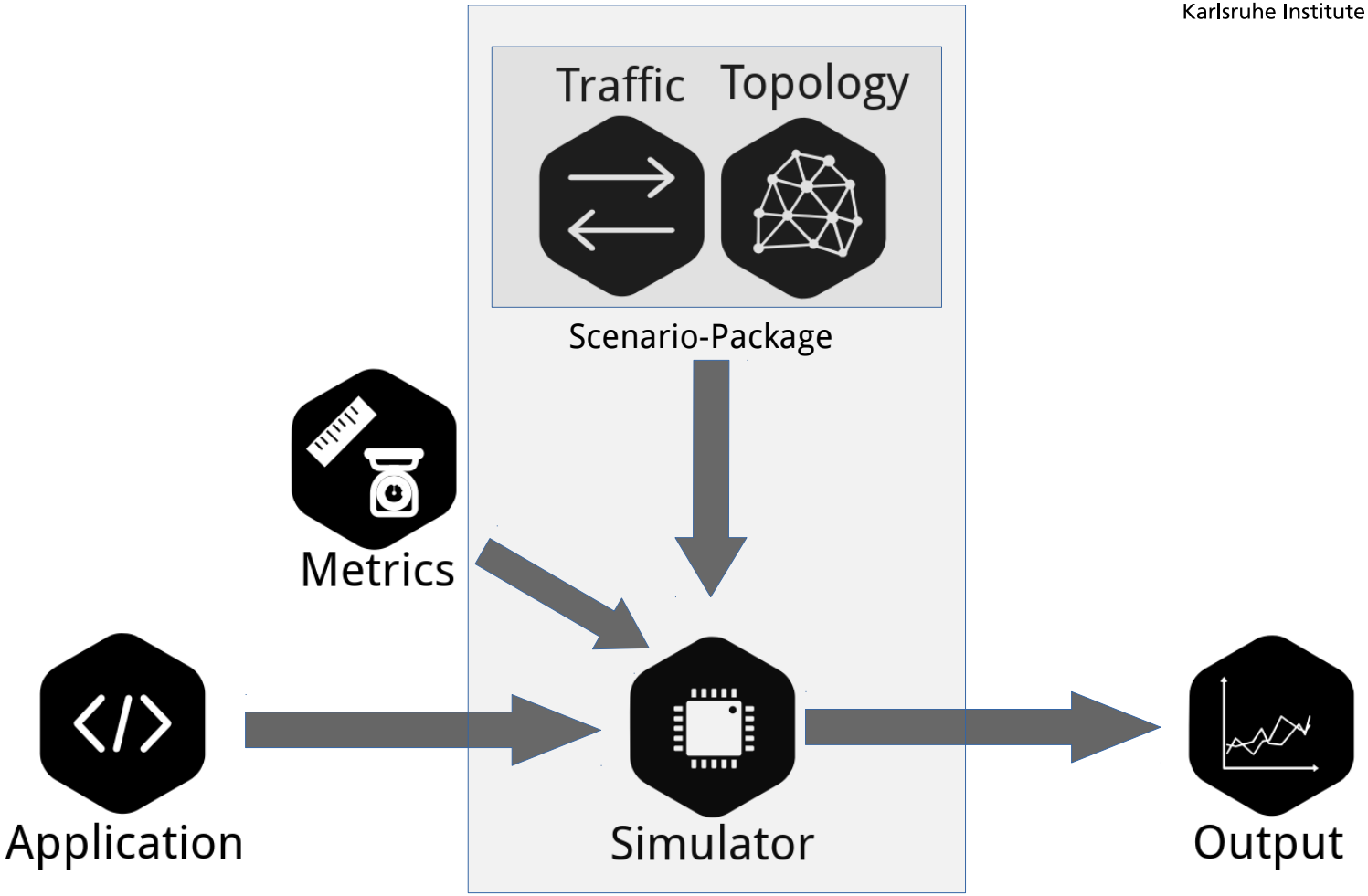
# SEED

Karlsruhe Institute of Technology

## Replace this:

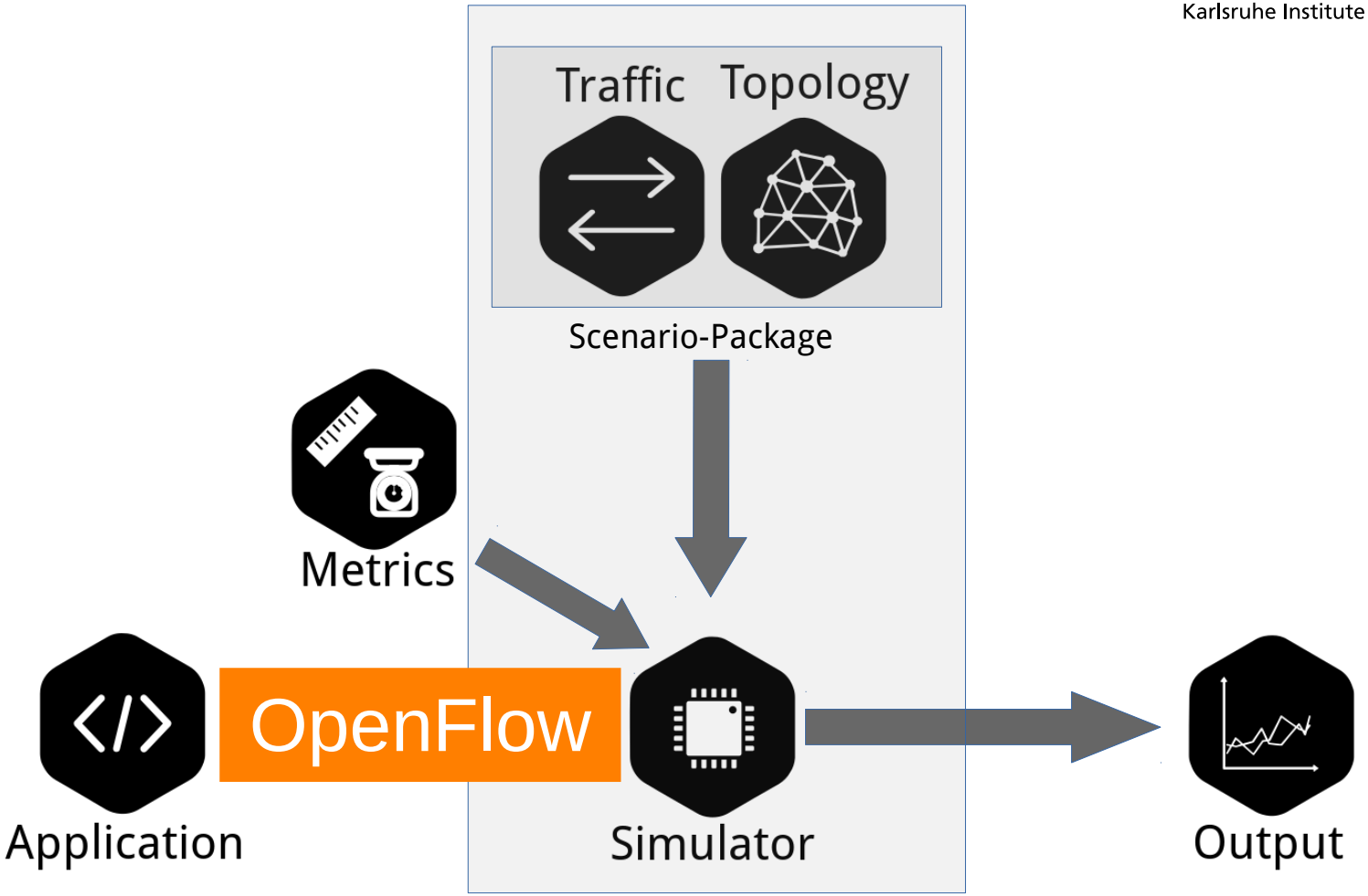


# SEED



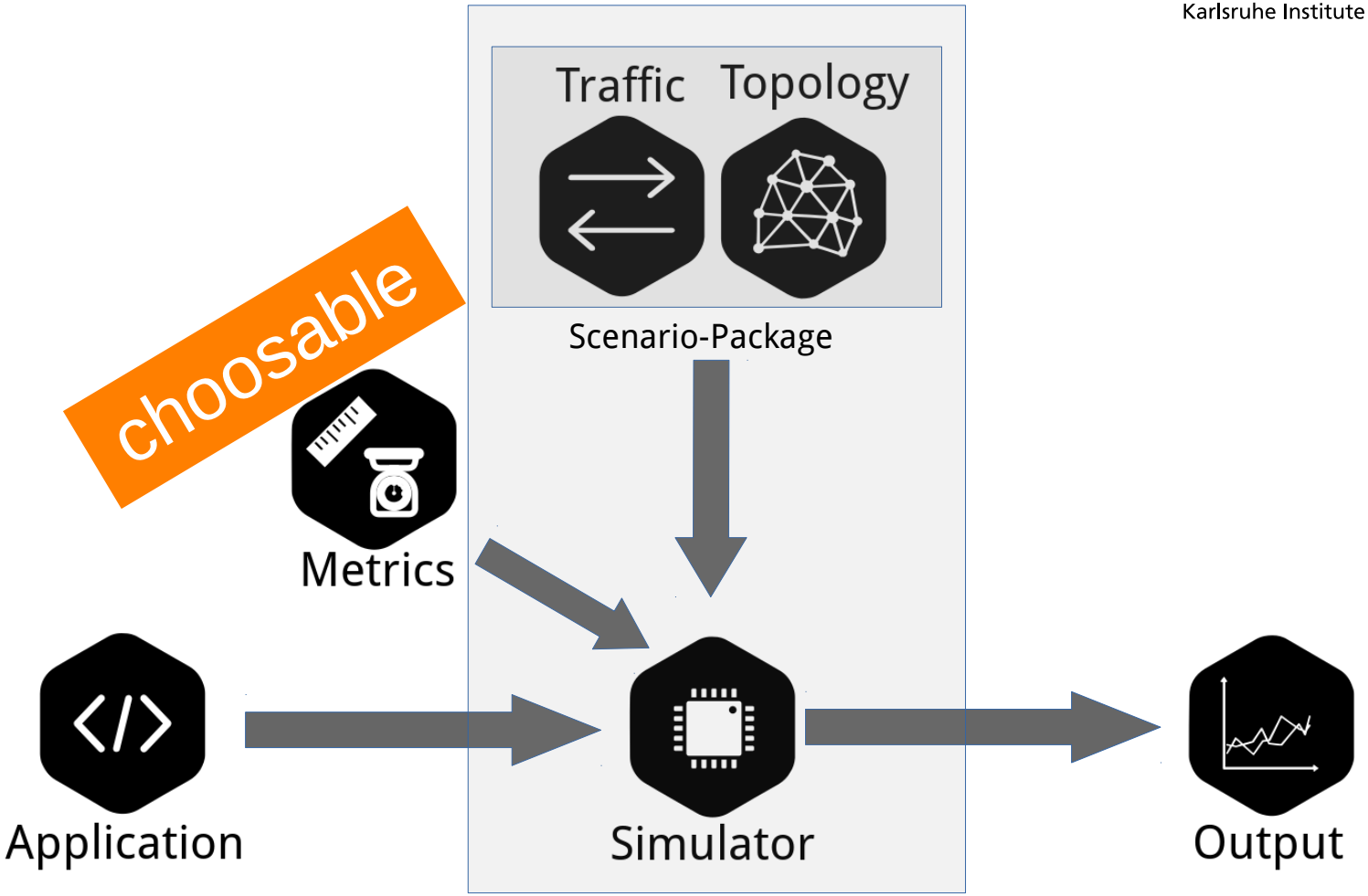

**Systematic Evaluation Environment for SDN-Applications**

# SEED



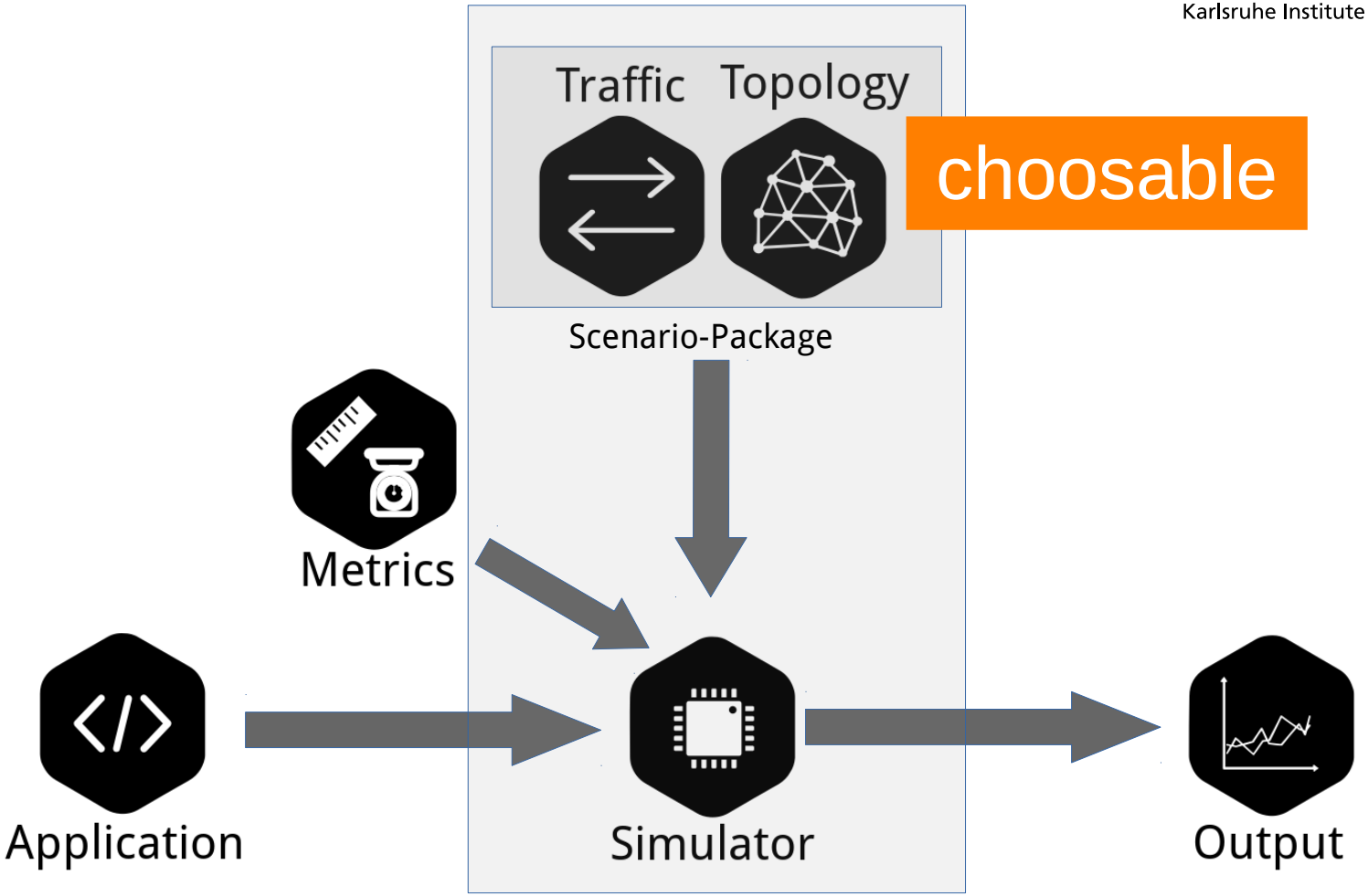

**Systematic Evaluation Environment for SDN-Applications**

# SEED



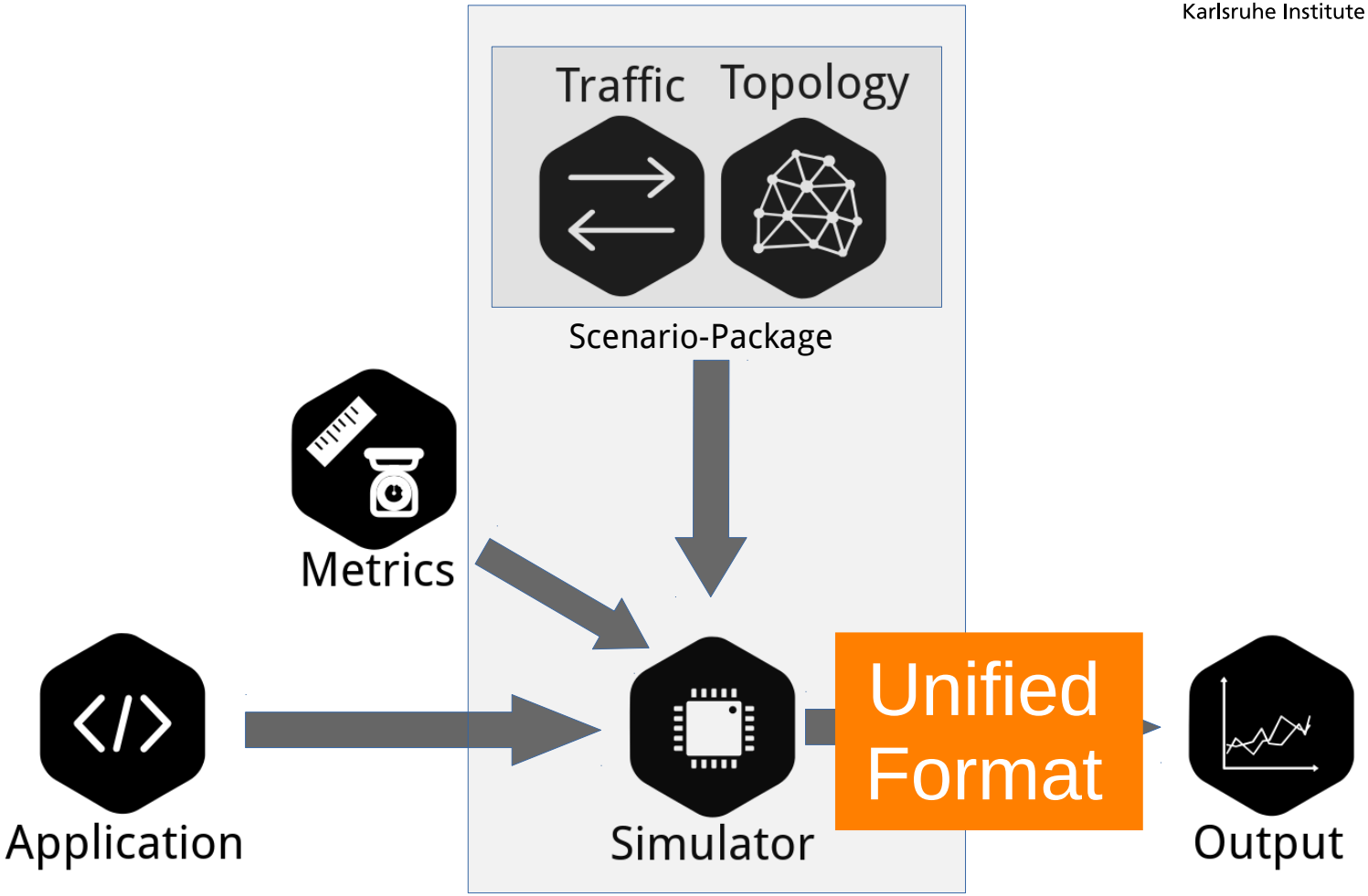

**Systematic Evaluation Environment for SDN-Applications**

# SEED



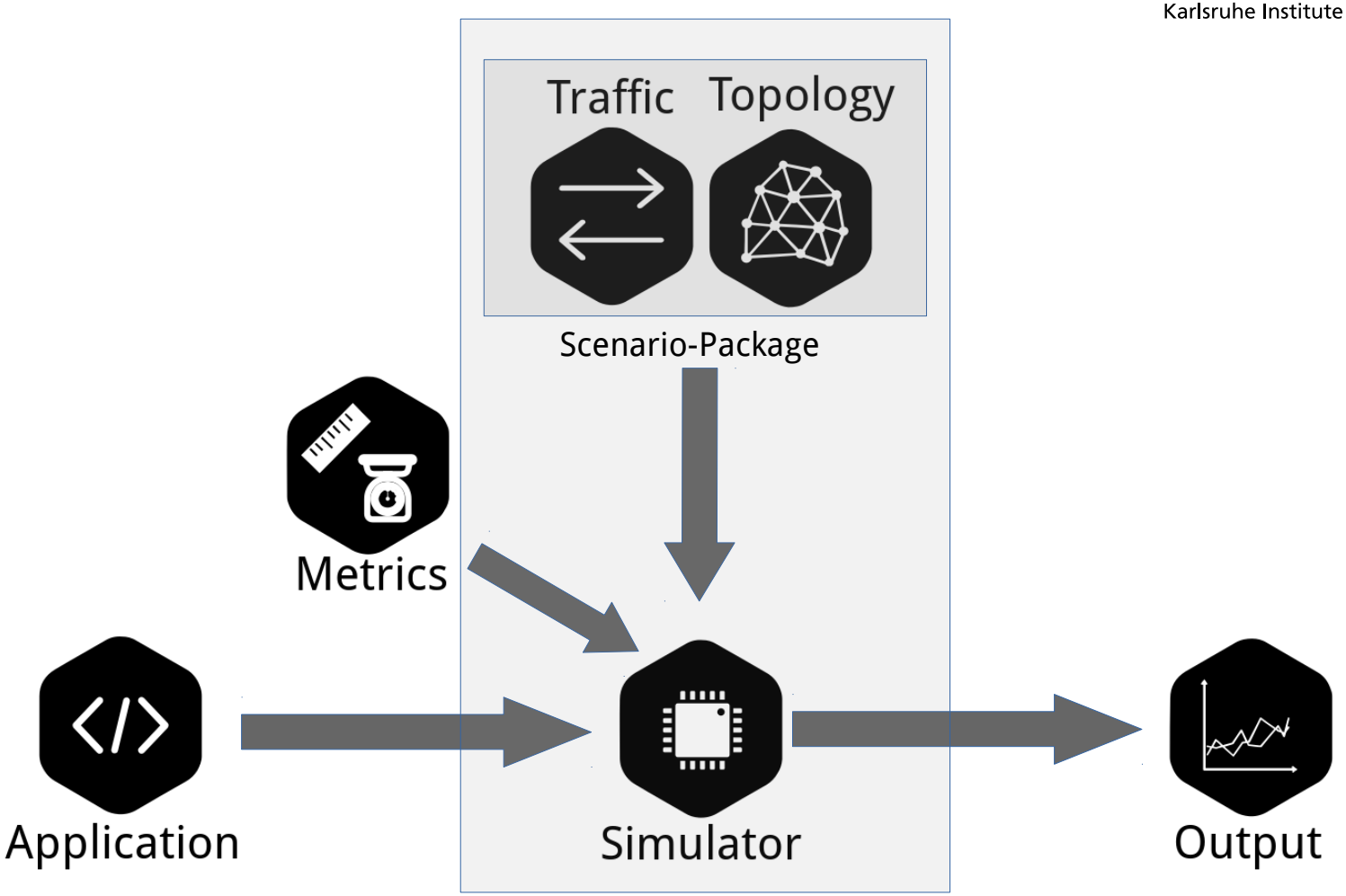

**Systematic Evaluation Environment for SDN-Applications**

# SEED



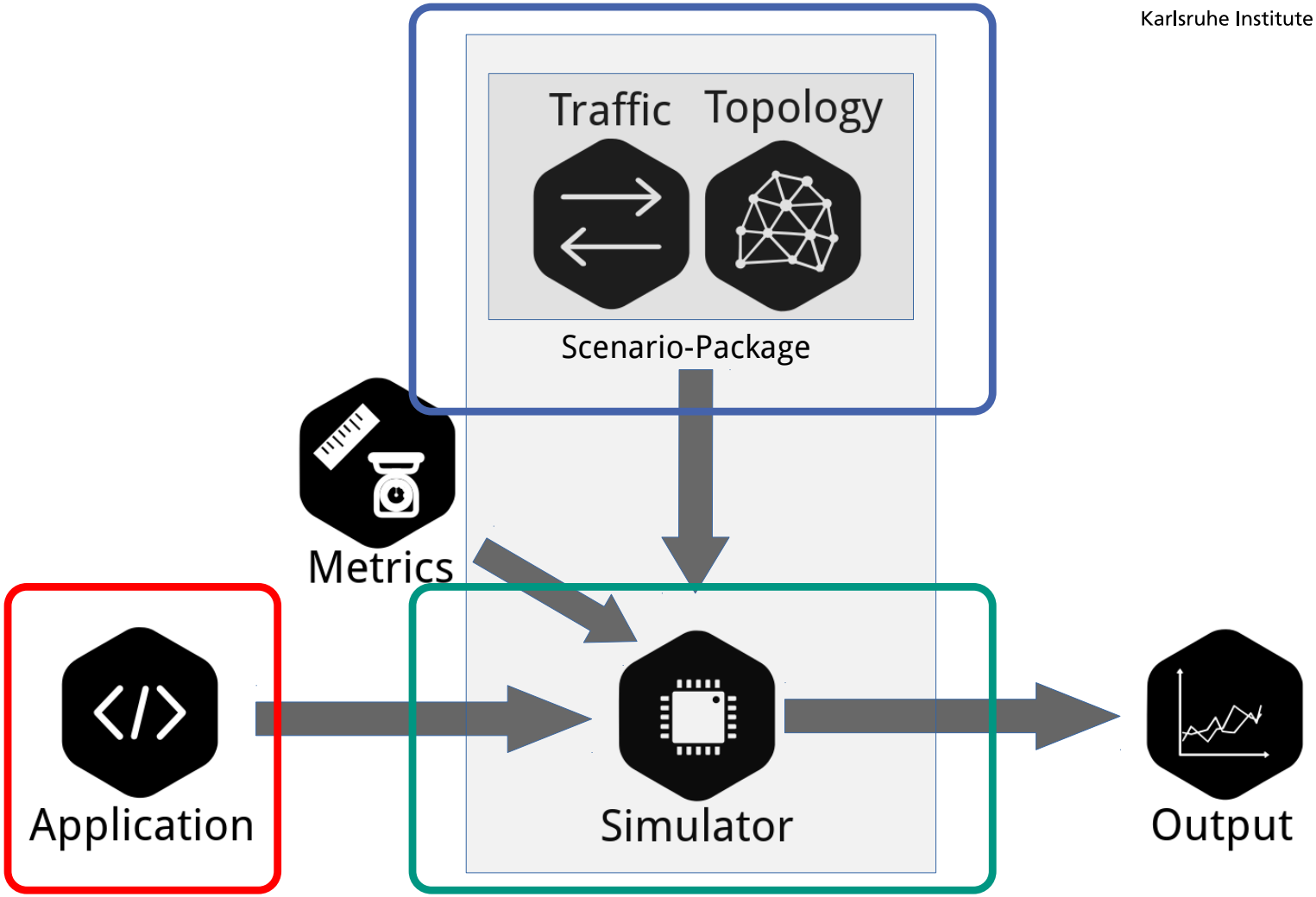

**Systematic Evaluation Environment for SDN-Applications**

# Components Exchangeable



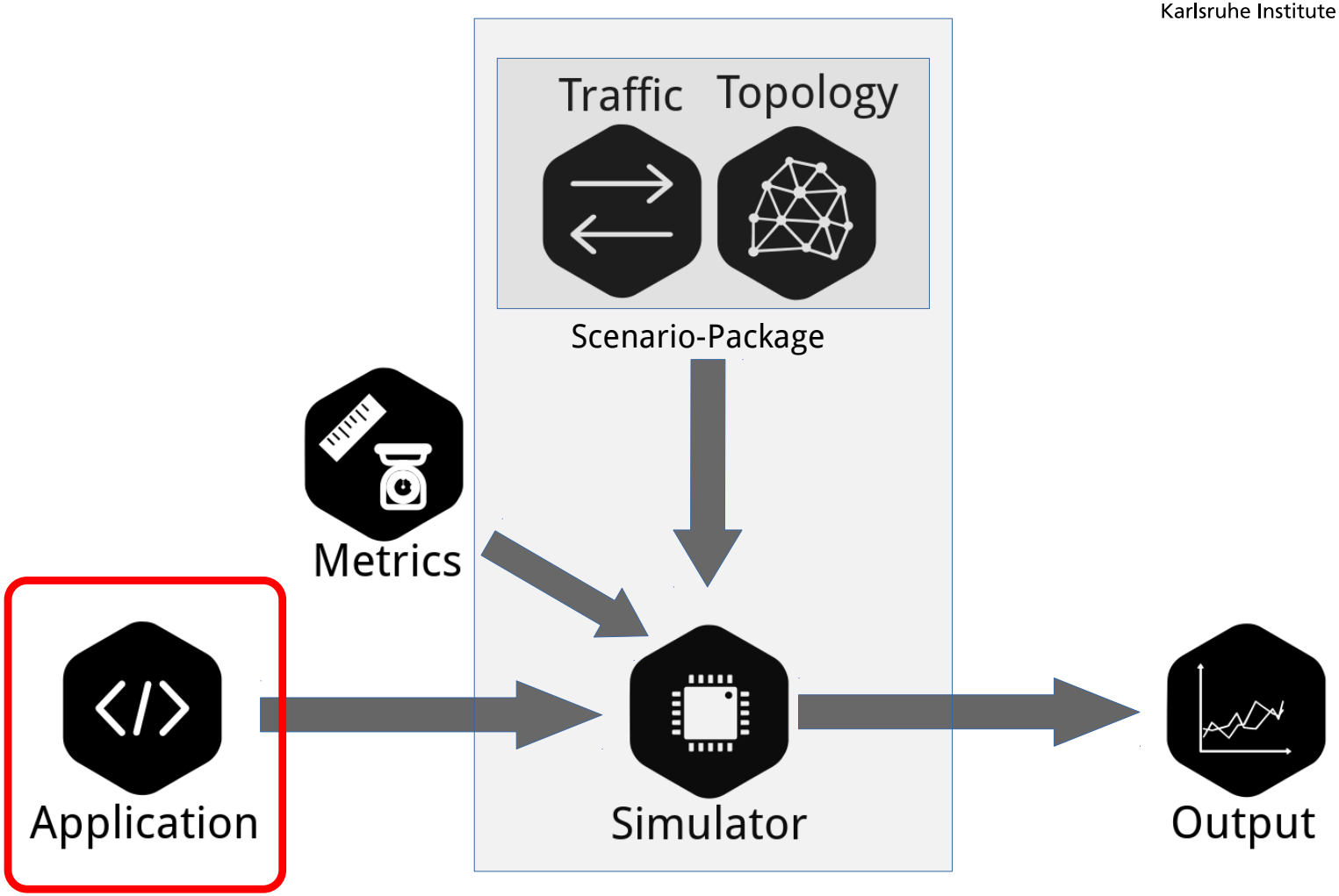

**Systematic Evaluation Environment for SDN-Applications**

# Components Exchangeable



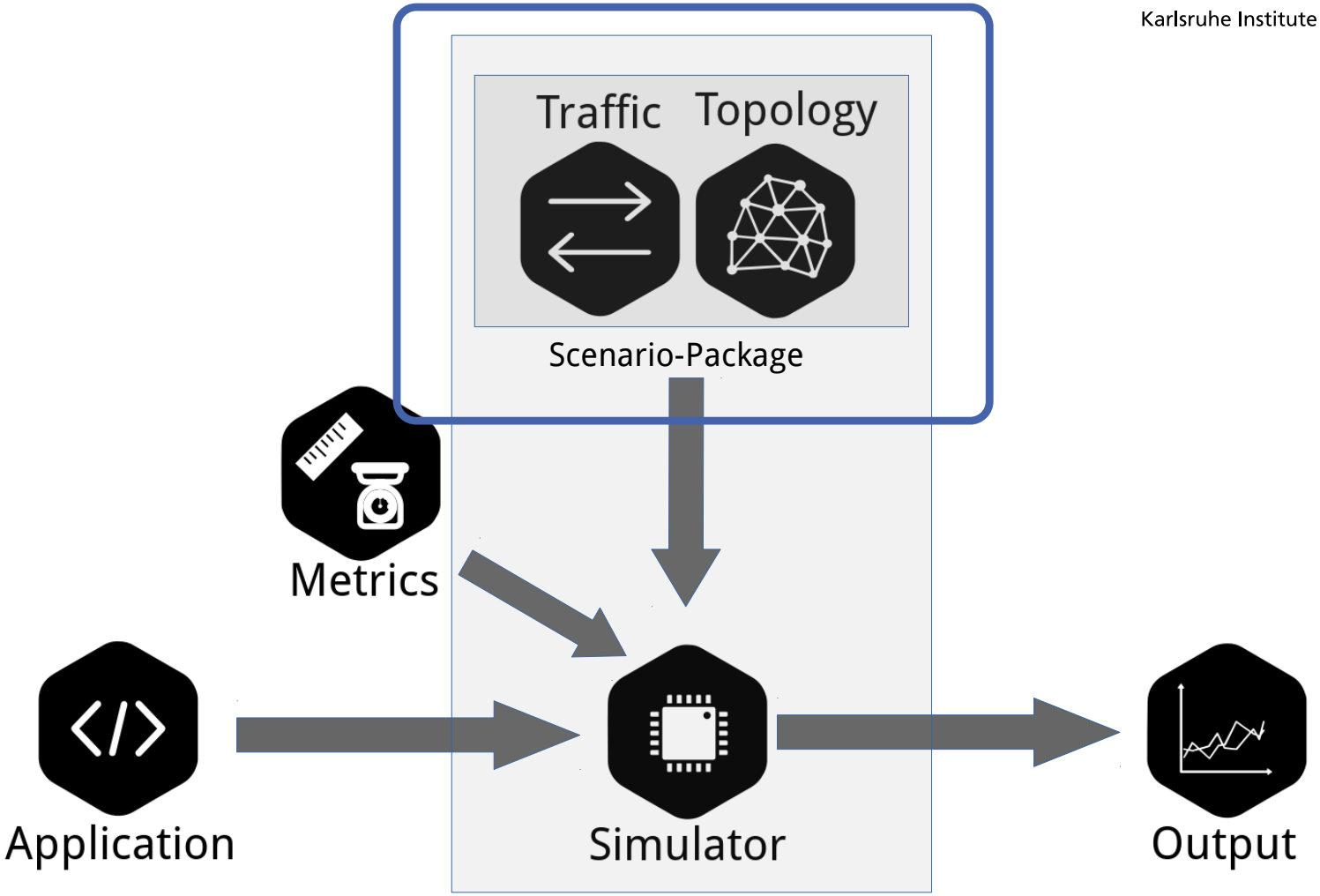

**Systematic Evaluation Environment for SDN-Applications**

# Components Exchangeable



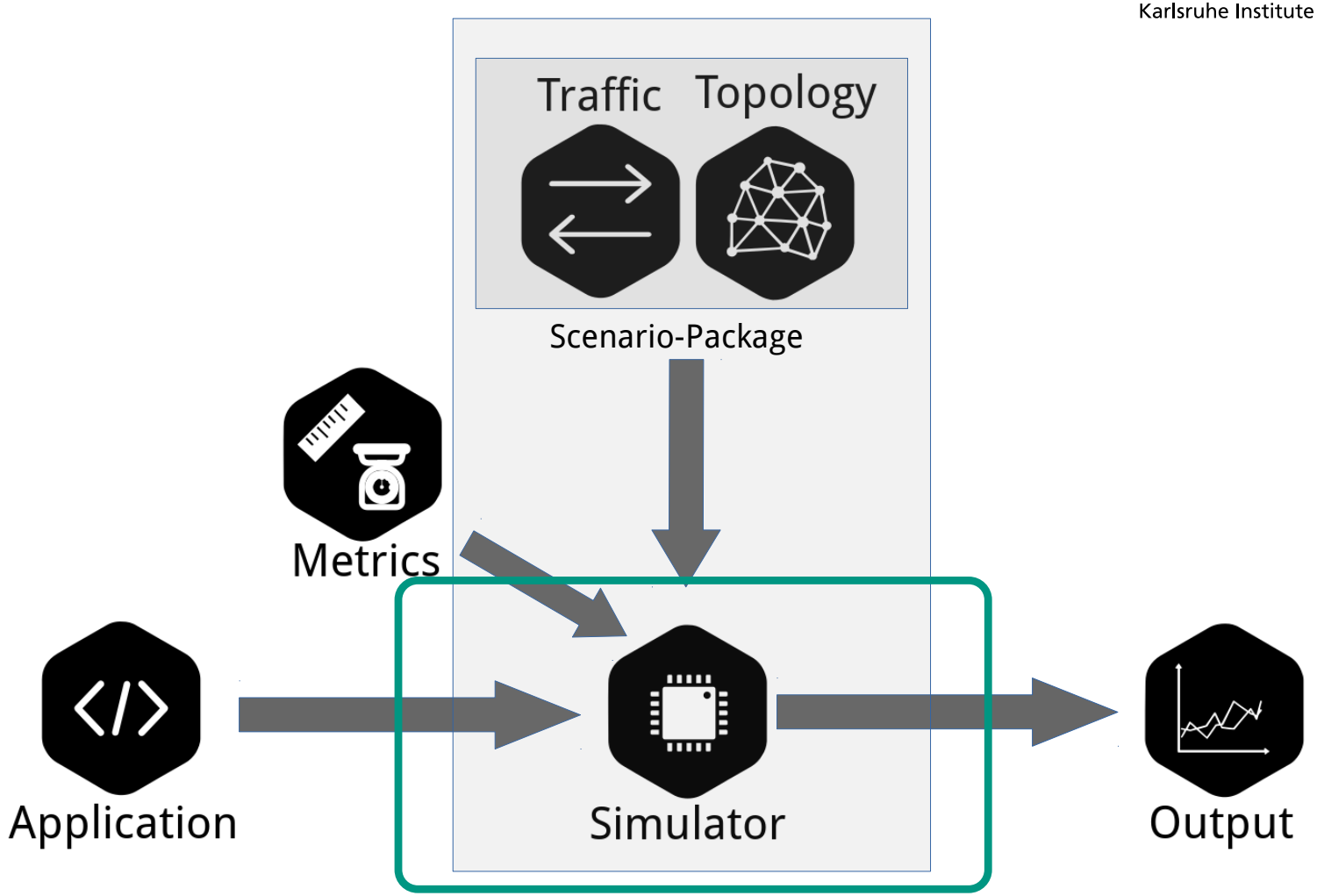

**Systematic Evaluation Environment for SDN-Applications**

# Components Exchangeable




**Systematic Evaluation Environment for SDN-Applications**

# Components Exchangeable



# Clear Experiment Description

- Enable reproducibility & sharing
  $\rightarrow$ one file describes complete evaluation setup

**Components**

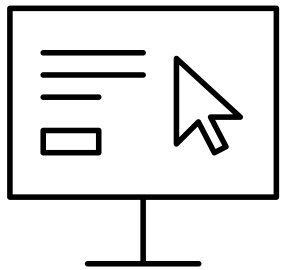
- Application(s)
- Scenario
- Simulator
- Parameters

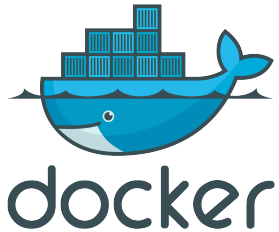
Experiment Description

# SEED-Frontend

**Unified starting point for experiments:**

- Preproccessing of configurations
- Initialization and start of
  - SDN-controller
  - Corresponding SDN-applications
  - Simulation environment
- Connection between components
- Docker Support: Faster setup

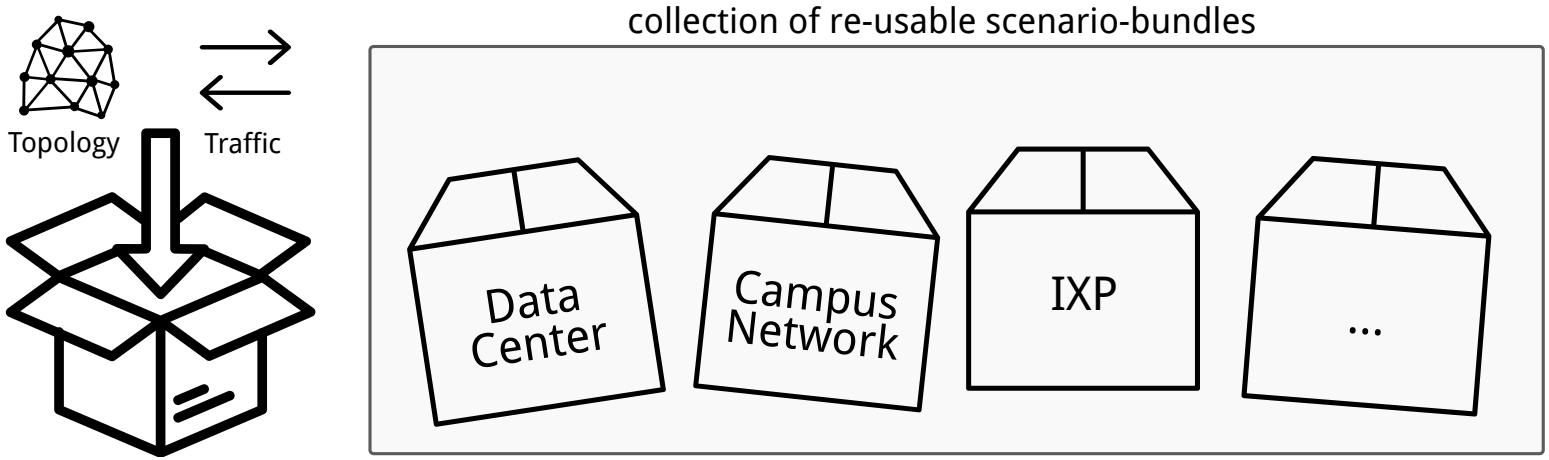




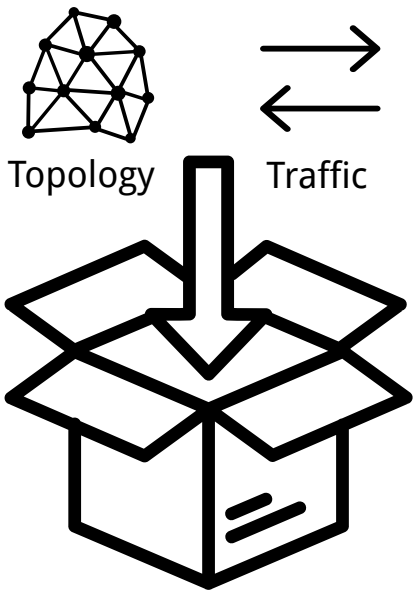
# Scenario-Bundles

Scenario (= Traffic + Topology)

- Enable uniform evaluation scenarios
- Fast experiment setup
- Easy sharing & reuse



# Format of Scenario-Bundles




## Describes complete scenario

- Topology
- Traffic

## Properties

- XML-based
- Addressing & grouping of network components
- Process-based traffic generation
- Integration of SDN-components

# Simulator-Adapters

Karlsruhe Institute of Technology

## Tasks

- Parsing of scenario-bundles
- Connection of SDN-components (via OpenFlow)
- Construction of the topology
- Execution of traffic & events

## Implementations

- mininet
- OMNeTT++
- ns-3

# Traditional Evaluation Process

Karlsruhe Institute of Technology

1. Choose simulator
2. Familiarize with simulator
3. Choose topology
   (investigation necessary)
4. Build topology in simulator
5. Design traffic model
   (investigation necessary)
6. Integrate traffic into simulator
7. Attach SDN-application
8. Execute simulator
9. Evaluate results

# Evaluation Process with SEED

Karlsruhe Institute of Technology

1. Choose simulator
2. Familiarize with simulator
3. Choose topology (investigation necessary)
4. Build topology in simulator
5. Design traffic model (investigation necessary)
6. Integrate traffic into simulator
7. Attach SDN-application
8. Execute simulator
9. Evaluate results

Choose/Provide

- Scenario-bundle
- Location of SDN-app

# Workflow

1. Choose simulator
2. Choose scenario-bundle
3. Provide path to own SDN-application
4. Execute SEED
5. Evaluate results

**Example usage:**

```
./seed -app app1 -scenario scenario1 -simulator sim1
./seed -app app2 -scenario scenario2 -simulator sim2
```

# Workflow

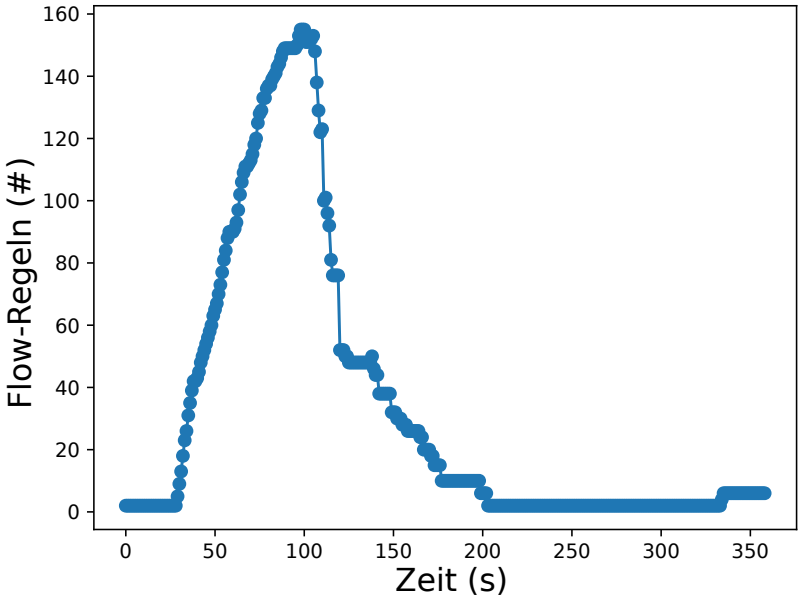
1. Choose simulator
2. Choose scenario-bundle
3. Provide path to own SDN-application
4. Execute SEED
5. Evaluate results

**Example usage:**

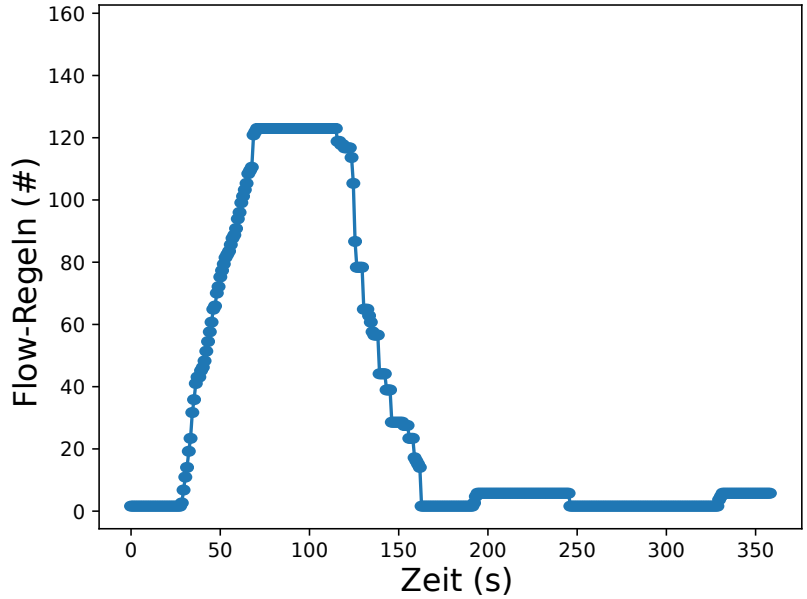
```
./seed -app app1 -scenario scenario1 -simulator sim1
./seed -app app2 -scenario scenario2 -simulator sim2
```

# Example Results

```
./seed -app switch -scenario datacenter -simulator mininet
./seed -app pbce -scenario datacenter -simulator mininet
```




Flowtable-Usage **without** PBCE




Flowtable-Usage **with** PBCE

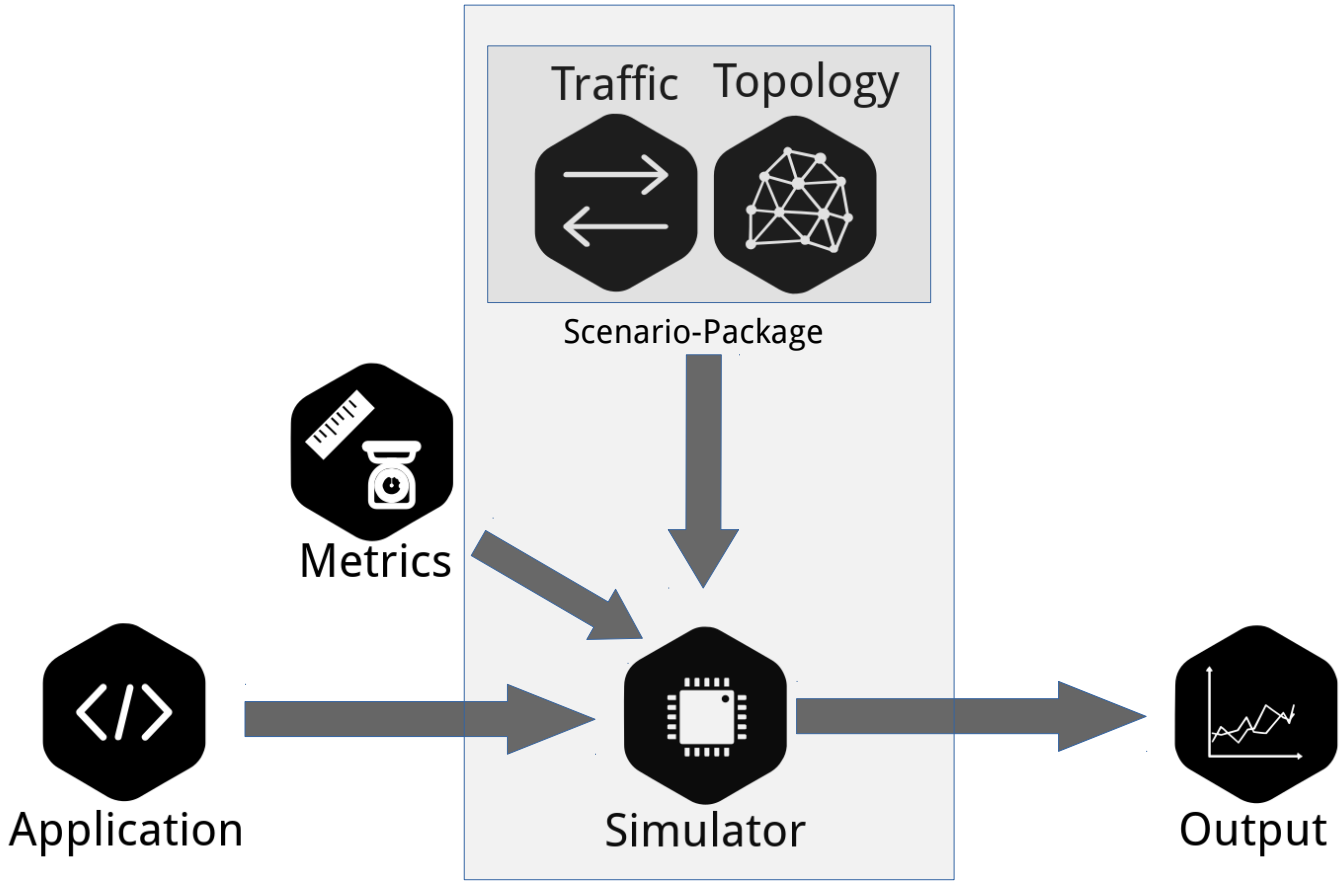
# Conclusion

SEED provides a systematic evaluation environment for SDN-applications.

It promotes and simplifies

- The evaluation process
- Reproducibility of evaluation results
- Comparability of evaluations

- Code will be made available as opensource
- Call for Participation: `https://git.scc.kit.edu/seed`




**Systematic Evaluation Environment for SDN-Applications**

# Sources I

Karlsruhe Institute of Technology

Icons: thenounproject.com