# OMNeT++ and mosaik: Enabling Simulation of Smart Grid Communications

Jens Dede, Koojana Kuladinithi, Anna Förster,
Okko Nannen[1], Sebastian Lehnhoff[1]

Sustainable Communication Networks
University of Bremen

OMNeT++ Summit
Thursday, 3rd September 2015

[1] R&D Division Energy, OFFIS, Germany

# Motivation

Power grid changes:

- Renewable energy resources
- Increasing decentralised power injection (households)
- Increasing number of electric vehicles

Communication networks challenges:

- Internet connectivity is available almost everywhere
- *Internet of Things*
- Real time control of the power grid

$\Rightarrow$ How to simulate power grid entities with communication capabilities?

# Main Challenges of Co-Simulation

- Most power grid simulators perform discrete **time** simulation
- Most network simulators perform discrete **event** simulation

| Discrete time simulation | $\Longleftrightarrow$ | Discrete event simulation |
|---|---|---|

$\Rightarrow$ How to synchronize both types of simulators?

# Existing Co-Simulation Frameworks

- **OPNET/Simulink testbed** for analysing effects of cyber intrusion on power grids [1]
- **FNCS**[1] is a power grid simulation framework. It has support for *ns-3*, *Gridlab-D* and *PowerFlow*, but can be extended for additional simulators [2, 3]
- **OMNeT++ & OpenDSS** are used to simulate a smart grid scenario [4, 5]
- **OMNeT++** can be used for analyzing real world measurements from the power grid and electric vehicles [6]

[1] Pronounced *phoenix*

# Existing Co-Simulation Frameworks

## Drawbacks

- Focussed on few simulators
- Use ns-3 or OPNET instead of OMNeT++
- OMNeT++ takes control of the complete simulation

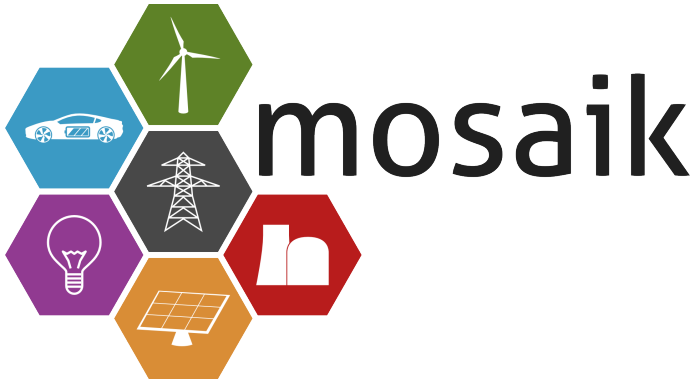
## Benefits

- How to combine discrete event and discrete time simulation
- How to synchronize both types of simulators

mosaik

# mosaik

- Flexible co-simulation framework for power grids
- Discrete time simulation
- OFFIS Energy and University of Oldenburg
- Used in SESA-Lab [1]
- License: LGPL





web: `mosaik.offis.de`

docs: `mosaik.readthedocs.org`

src: `bitbucket.org/mosaik`

[1] Smart Energy Simulation and Automation Laboratory

# Advantages of mosaik

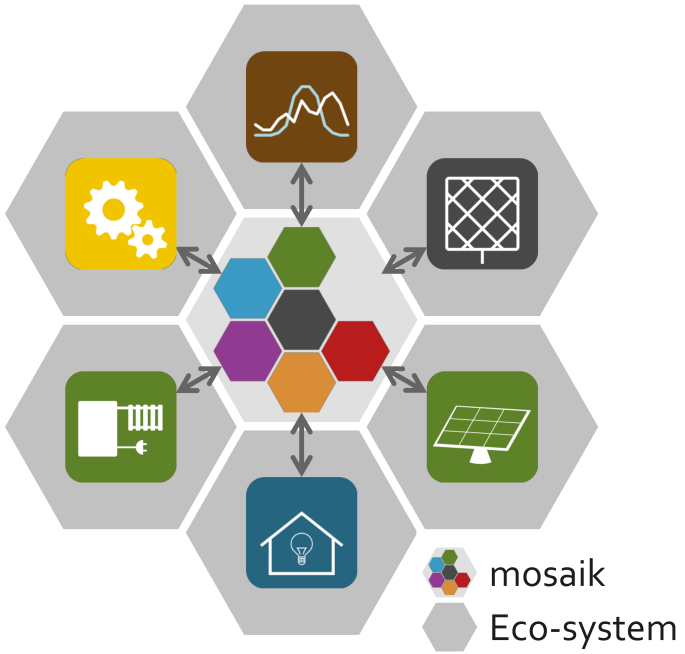
- Supports a remarkable number of simulators and simulator types
  - Environment and weather conditions
  - Energy markets
  - Information and communication systems
- Visualisation of results and simulator interactions
- Business services available by OFFIS
- Support for simulation cluster
- Well documented, example scenario available online
- Open Source

$\Rightarrow$ mosaik assumes a perfect communication link between the entities

# mosaik System Architecture

Tasks of mosaik:

- Offers a generic API for simulators
- Orchestrate between different simulators
- Controls the simulators




How to embed a simulator?

- **Python** module: `package.module:SimClass`
- **Command line**: `java -jar simB.jar %(addr)s`
- **Socket**: `targethost:5678`

# mosaik API

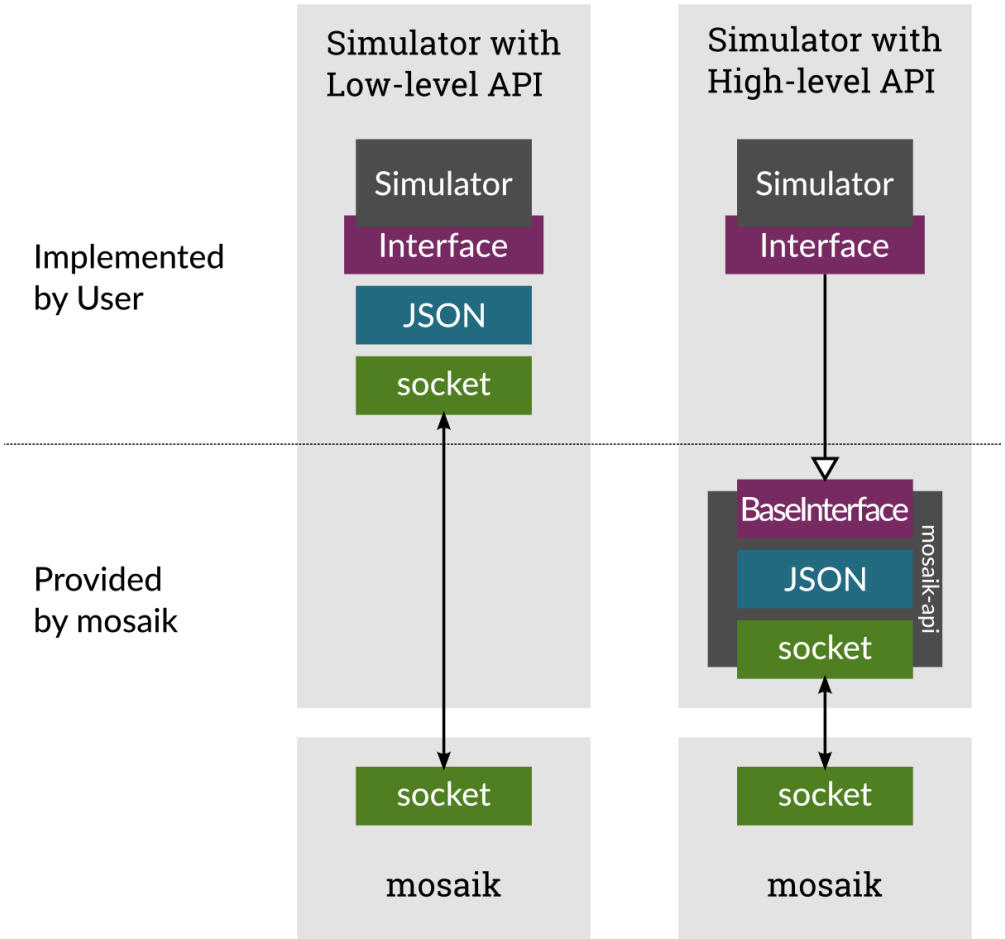



Figure 1: mosaik APIs

# mosaik API

The mosaik API performs the following API calls:

- `init()`
- `create()`
- `setup_done()` (opt.)
- `step()`
- `get_data()`
- `stop()`

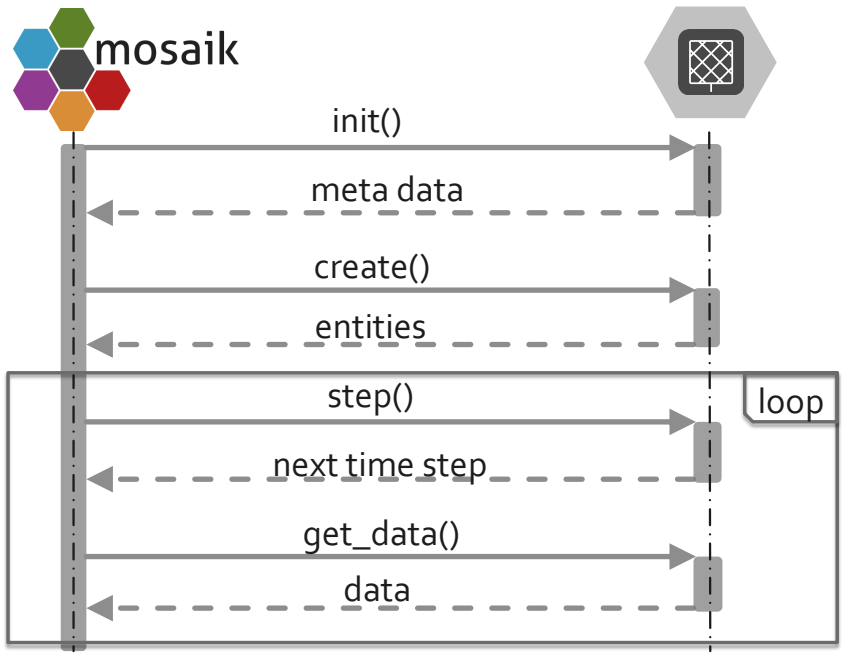



Figure 2: mosaik API calls

# mosaik Visualization Tool

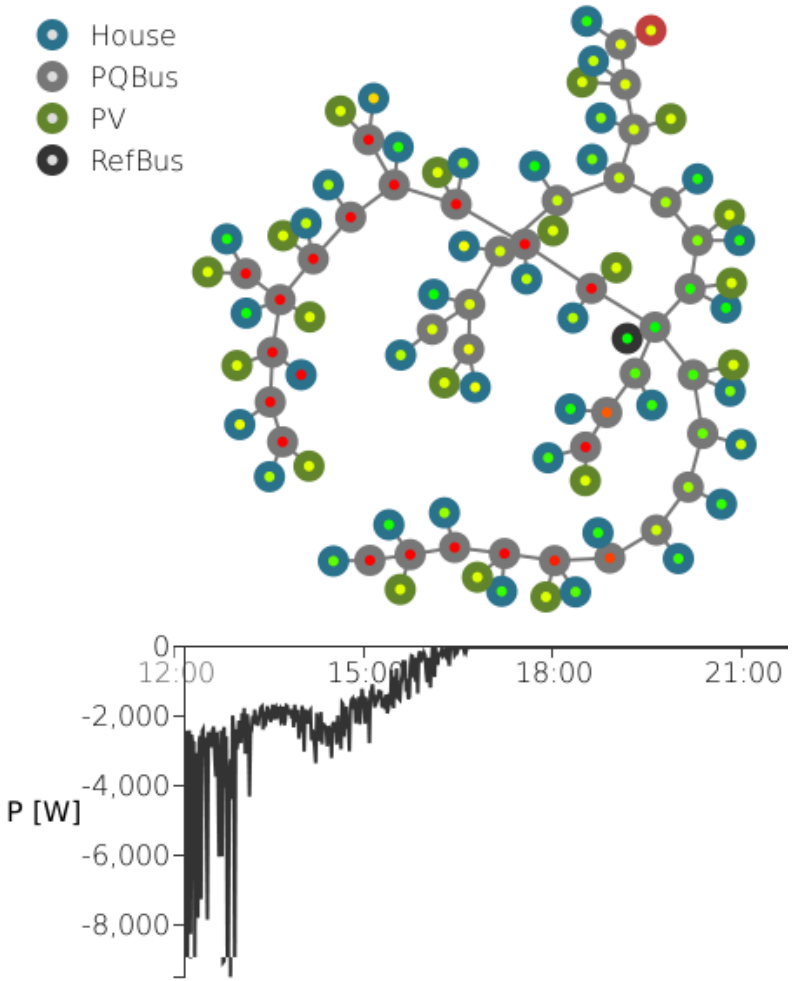



Figure 3: mosaik grid visualization

# OMNeT++

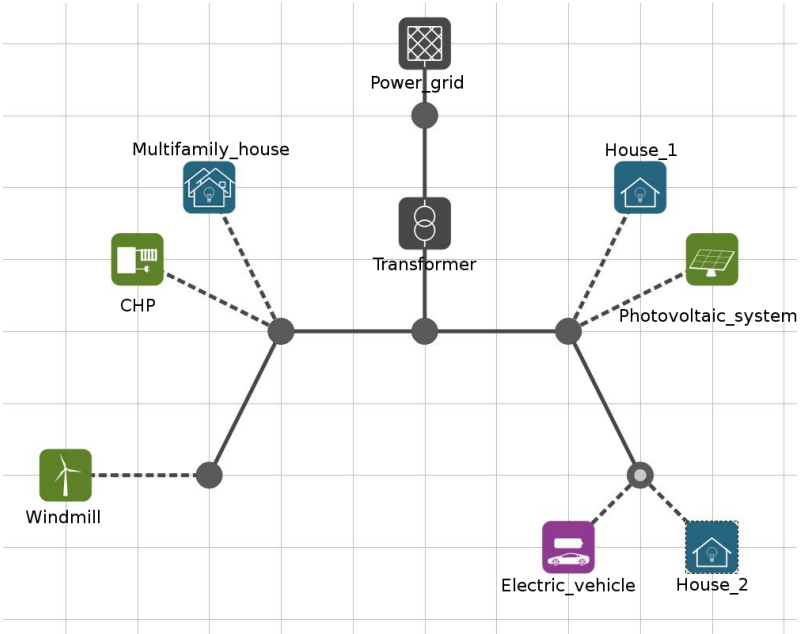
# Co-Simulation Approaches for OMNeT++

- Most cases: OMNeT++ controls the simulation
- No way to control OMNeT++ externally
  i.e. execute simulation event by event

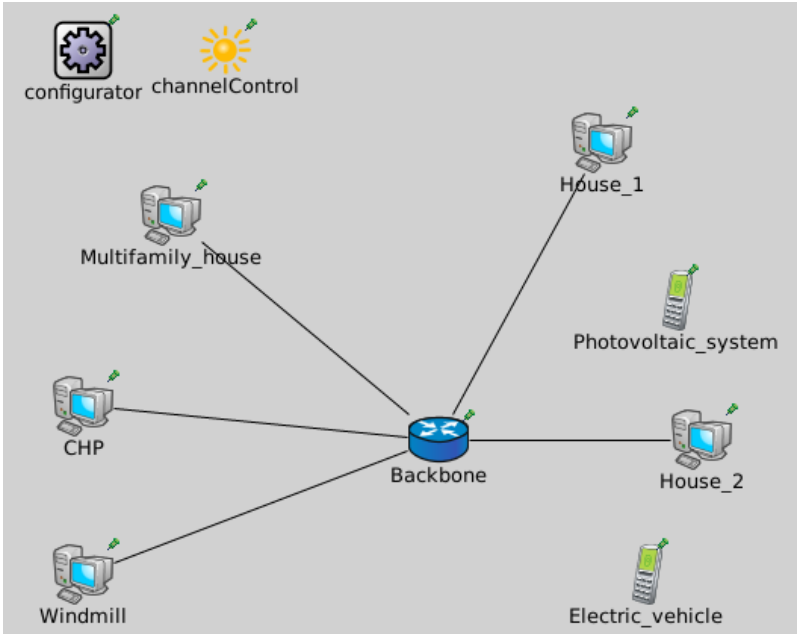
Due to the modularized architecture of OMNeT++:
Only an implementation task

# Combine OMNeT++ & mosaik

# Preliminary System Architecture




(a) The mosaik representation




(b) The OMNeT++ representation

Figure 4: An example smart grid scenario

# Preliminary System Architecture

Steps towards federated simulation:

1. Extract communication network from mosaik
2. Full external control of OMNeT++ and implementation of the mosaik API
3. Improve performance of overall simulation

# Extract Communication Network

mosaik uses `json` to describe simulation setup:

- Can be extended easily to describe communication links
- Need to be converted to a `.ned`-file

# Adaptations to OMNeT++

OMNeT++ allows embedded simulations: **no external control**

We need to do the following:

- Initialize the simulation
- Run events one by one, pass results to mosaik
- Insert events externally (i.e. streams, packets)
- forward errors to mosaik
- Finish the simulation

# Adaptations to OMNeT++

How to achieve that?

⇒ Implement functions similar to the existing `simulate()`:

- `simulate_with_control()`
- `step_one_event()`
- `simulate_until(<timestamp>)`

Why not perform a realtime simulation?

- Simulation can take a long time (months or years)
- Performing a realtime simulation with complex scenarios not applicable

# Improve Performance

Event-based synchronization is inefficient

- Pass results only on application layer
- Multiple event simulation
- Identify independent communication flows
- Use existing parallelization techniques in OMNeT++

# Conclusion and Future Work

*Integrating OMNeT++ in mosaik will result in a power grid simulator capable to simulate not only energy, but also the communication links of nowadays and future smart grid scenarios.*

Main tasks:

- Enable external control of OMNeT++
- Implement mosaik API
- Convert mosaik meta data to `.ned`-file
- Improve performance

# References I


[1] M. A. H. Sadi, M. H. Ali, D. Dasgupta, and R. K. Abercrombie, "Opnet/simulink based testbed for disturbance detection in the smart grid," in *Proceedings of the 10th Annual Cyber and Information Security Research Conference*, ser. CISR '15. New York, NY, USA: ACM, 2015, pp. 17:1–17:4. [Online]. Available: http://doi.acm.org/10.1145/2746266.2746283

[2] S. Ciraci, J. Daily, J. Fuller, A. Fisher, L. Marinovici, and K. Agarwal, "Fncs: A framework for power system and communication networks co-simulation," in *Proceedings of the Symposium on Theory of Modeling & Simulation - DEVS Integrative*, ser. DEVS '14. San Diego, CA, USA: Society for Computer Simulation International, 2014, pp. 36:1–36:8. [Online]. Available: http://dl.acm.org/citation.cfm?id=2665008.2665044

[3] S. Ciraci, J. Daily, K. Agarwal, J. Fuller, L. Marinovici, and A. Fisher, "Synchronization algorithms for co-simulation of power grid and communication networks," in *Modelling, Analysis Simulation of Computer and Telecommunication Systems (MASCOTS), 2014 IEEE 22nd International Symposium on*, Sept 2014, pp. 355–364.

# References II


[4] M. Lévesque, D. Q. Xu, G. Joós, and M. Maier, "Communications and power distribution network co-simulation for multidisciplinary smart grid experimentations," in *Proceedings of the 45th Annual Simulation Symposium*, ser. ANSS '12. San Diego, CA, USA: Society for Computer Simulation International, 2012, pp. 2:1–2:7. [Online]. Available: http://dl.acm.org/citation.cfm?id=2331751.2331753

[5] D. Bhor, K. Angappan, and K. Sivalingam, "A co-simulation framework for smart grid wide-area monitoring networks," in *Communication Systems and Networks (COMSNETS), 2014 Sixth International Conference on*, Jan 2014, pp. 1–8.

[6] S. Bocker, C. Lewandowski, C. Wietfeld, T. Schluter, and C. Rehtanz, "Ict based performance evaluation of control reserve provision using electric vehicles," in *Innovative Smart Grid Technologies Conference Europe (ISGT-Europe), 2014 IEEE PES*, Oct 2014, pp. 1–6.

[7] M. Buscher, A. Claassen, M. Kube, S. Lehnhoff, K. Piech, S. Rohjans, S. Scherfke, C. Steinbrink, J. Velasquez, F. Tempez, and Y. Bouzid, "Integrated smart grid simulations for generic automation architectures with rt-lab and mosaik," in *Smart Grid Communications (SmartGridComm), 2014 IEEE International Conference on*, Nov 2014, pp. 194–199.