Analyzing the behavior of communication protocols is an important task for validation and to find opportunities for improvement. For wireless sensor networks (WSN), protocols are tested in simulation and on testbeds consisting of real hardware devices (nodes). During execution - in simulation or on real hardware - the nodes create detailed logs that can be used to analyze the protocol in hindsight. The logs contain the output information of each node along with a timestamp that allows the entries of the different nodes to be ordered in time. Output information include received/transmitted packets, reception failures, or other relevant metrics. In general, these logs can become large such that manual analysis is very time-consuming and complicated. A tool that provides a visual overview of the log to quickly and better understand the interaction between nodes and their behavior over time would be a valuable addition to the protocol analysis.

The goal of this project is to develop a visualization tool in Python that uses log files as input. In a first step, the tool should provide an overview of the network topology and visualize the packet exchange between the nodes over time (see screenshot). This is a general task, since the exchange of packets is the basis of all communication protocols. An important part of this task is to design a suitable interface between the protocol logs and the visualization tool.

In order to gain deeper insights into the protocol behavior, it is necessary to extend the tool in a second step to visualize protocol-specific information as well. One example is the propagation of data in the network, which is a metric over a time interval. Another example are information about nodes at a certain point in time, for instance which nodes were observed in the neighborhood, or the amount of data already collected. Some of these protocol-specific information can be read directly from the log, while others must first be derived from the information in the log. In the latter case, the tool should provide extension points so that user-defined metrics can be computed and visualized.

As a starting point, the student can use a basic version of this tool written in Python. The creation of log files is not part of the project, and the student is given existing logs that contain all relevant information. Depending on the targeted project proposal type, the scope and complexity of the task can be varied.

Requirements
- Good coding skills in Python.
- Experience with Qt is a nice-to-have but not required.

Contact
- Fabian Mager, fabian.mager@tu-dresden.de
- Prof. Marco Zimmerling, zimmerling@cs.uni-freiburg.de