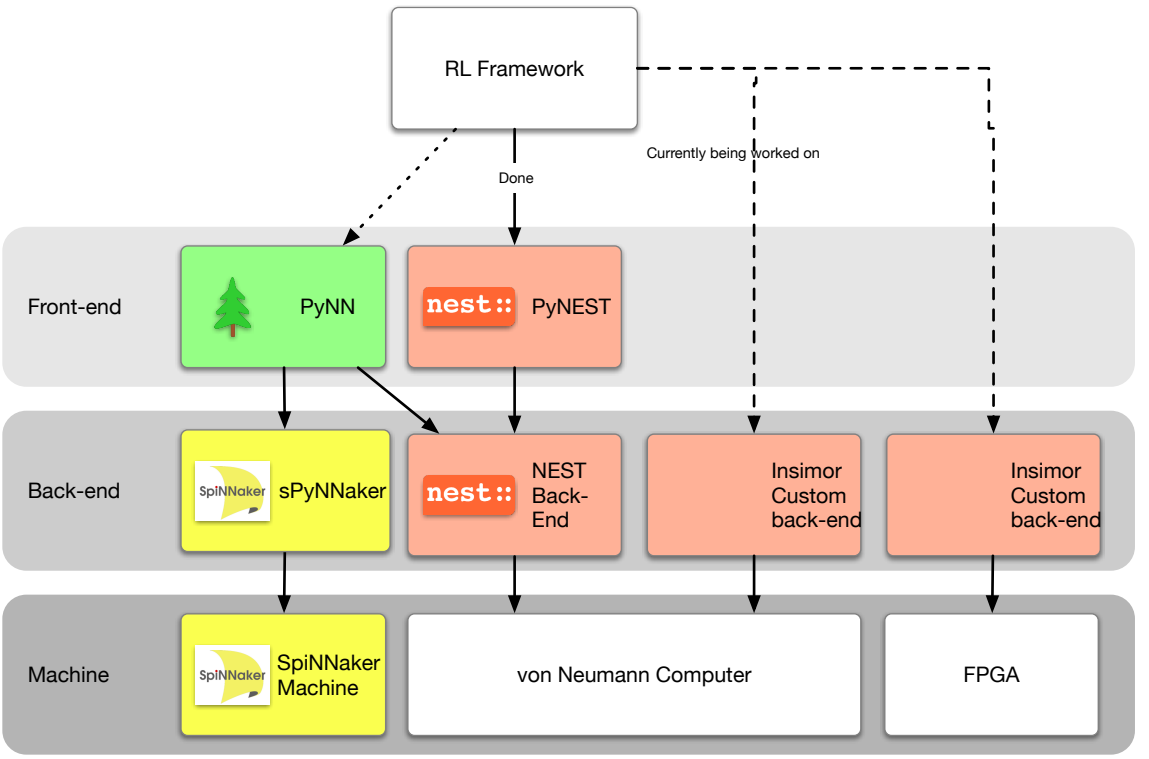
Learning on Neuromorphic Hardware: A path

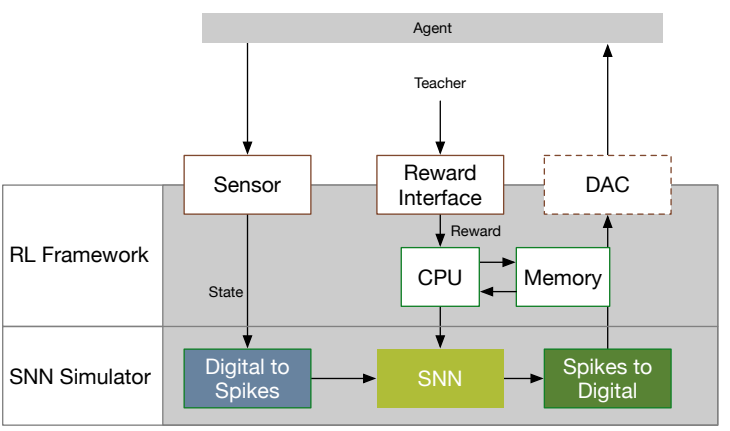
Benedikt S. Vogler, 2021

In my master thesis I wrote a Reinforcement Learning (RL) framework and made the Spiking Neural Network (SNN) run on the neural simulator back-end Nest. The algorithm of interest is a dopamine based actor-learning in a spiking neural network. The algorithm works for basic problems but it needs more research to find the mechanisms and neural architectures for SNN RL. The NEST simulator is interfaced with Python (PyNest). The problem is that PyNest is not optimized for short feedback loops. This caused trainings to need around 8 hours for simple problems unable to run in real-time. Therefore I am currently rewriting a custom back-end in multithreaded C++ with a Python Interface, called Insimor (for **i**nput **sim**ulation **o**utput **r**eward). This is preparing the next step: fully neuromorphic hardware.

Architecture



Framework



Needed steps

- Simplified numeric algorithm in Python
- SNN algorithm in PyNest
- Simplified numeric algorithm in Insimor
- SNN algorithm in Insimor
- Run training on real robot agent
- Implement Insimor back-end on FPGA

The Insimor back-end allows the computation of the three components in the SNN simulator in parallel on weak hardware as the Raspberry Pi. By using FPGAS each neuronal computation can happen in hardware, all in parallel, thus boosting the performance by a hundredfold and allowing further scaling on more expensive or specialized (neuromorphic) hardware.

Robot agent work in progress