Insite Pipeline

A Pipeline Enabling In-Transit Processing for Arbor, NEST and TVB

Marcel Krüger[1] - Tim Gerrits[1] - Torsten Kuhlen[1] - Benjamin Weyers[2]

[1] Visual Computing Institute, RWTH Aachen University, Aachen, German

[2] Human-Computer Interaction, University of Trier, Trier, Germany

Human Brain Project

MOTIVATION

Neuronal network simulations has advanced steadily to larger and more complex models

- **Major issues and challenges: Scaling**
  - Demanding amount of produced data and simulation runtime
  - Limited disk space & long waiting time
- **Standard solution: In-transit approaches** [8]
  - Allows access to data while simulation runs
  - Parallel processing outside the simulation
  - Early insights into results → allows early termination
- **Problem of off-the-shelf in-transit solutions: Integration into workflows**
  - Involved integration into workflows
  - Many dependencies or complex data structures
- **Consequences:**
  - Important features are not accessible to all users
  - Domain scientists cannot leverage in-transit processing

OUR OBJECTIVE

Providing easy-to-use and easy-to-integrate in-transit access to simulation data of multiscale simulations conducted with TVB [6], NEST [4], and Arbor[5].

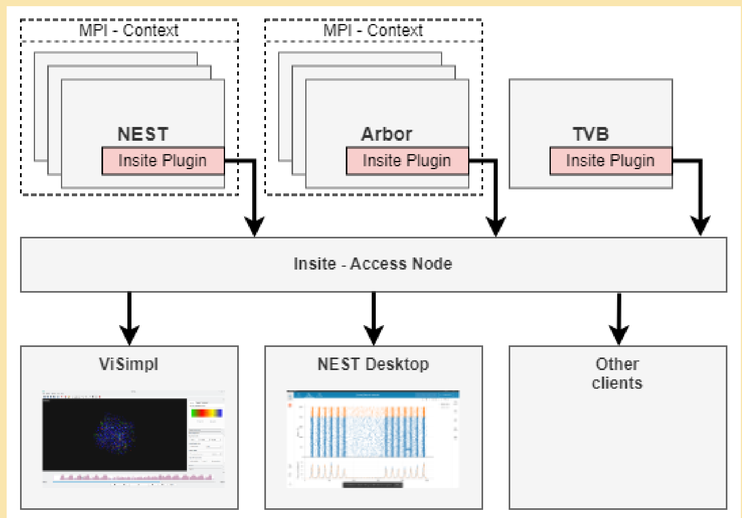

Figure 1. The Insite Pipeline data flow. Data from multiple simulators is available via the Access Node. Multiple clients can access data in parallel via broadly available protocols and data formats.

METHODS

```python
import requests
import json

def get_data(url, query_params):
  response = requests.get(url)
  json_sim_time = json.loads(response, params=query_params)
  return json_sim_time

sim_info = "http://localhost:52056/nest/simulationTimeInfo"
spike_url = "http://localhost:52056/nest/spikes"

while True:
  if get_data(sim_info)["current"]==0:
    continue
  if get_data(sim_info)["end"]==get_data(sim_info)["current"]:
    break

  print(get_data_as_json(spike_url,{"nodeIds":[10,2345]}))
```

Figure 2. Simple client to access in-transit data from NEST via Python. The client uses the requests API to get the simulation time and print spike information for nodes with id 10 and 2345

- **Insite: Interface for easy integration into existing workflows and tools**
  - Modular and tiered architecture
- **Simulator modules**
  - Provided for TVB, NEST, and Arbor
  - Responsibility: collecting raw data from the simulation & providing it to the access node
  - Unintrusive and easy for all users to integrate into existing simulation scripts
- **Access node**
  - Single point of contact for users
  - Provides the data of all simulators and simulator instances from a single source
- **Standard protocols and data formats**

→ Accessible in various programming languages and technologies (many libraries are available)

- **Data access via two paradigms**
  - Push-oriented via WebSockets: New data pushed into user's application when new data is available
  - Pull-based via HTTP REST API: Query data on demand
- **Data returned**
  - JSON: human-readable representation with broad support
  - Flatbuffer: more performant alternative (binary encoding) to JSON
- **The API provides a variety of parameters**
  - For filtering the data and accessing as easy as possible for further processing

RESULTS & DISCUSSION

- **Ease-of-integration and ease-of-use**
  - Accessible to many developers and users
  - Successfully integrated into
    - ViSimpl Visualization Tool [3]
    - NEST Desktop[7]
  - Insite's design allowed to add it to their desktop and web applications with ease
  - Enabling in-transit capabilities & early feedback on their simulations
- **In the context of HBP**
  - Emerging eco-system of web-based solutions
  - Insite provides advantages over classical in-situ/in-transit approaches
  - Modular architecture allows extending the access node → providing more data, e.g., metrics or pre-processed data from other sources
  - Insite enriches the capabilities of the computational neuroscience community

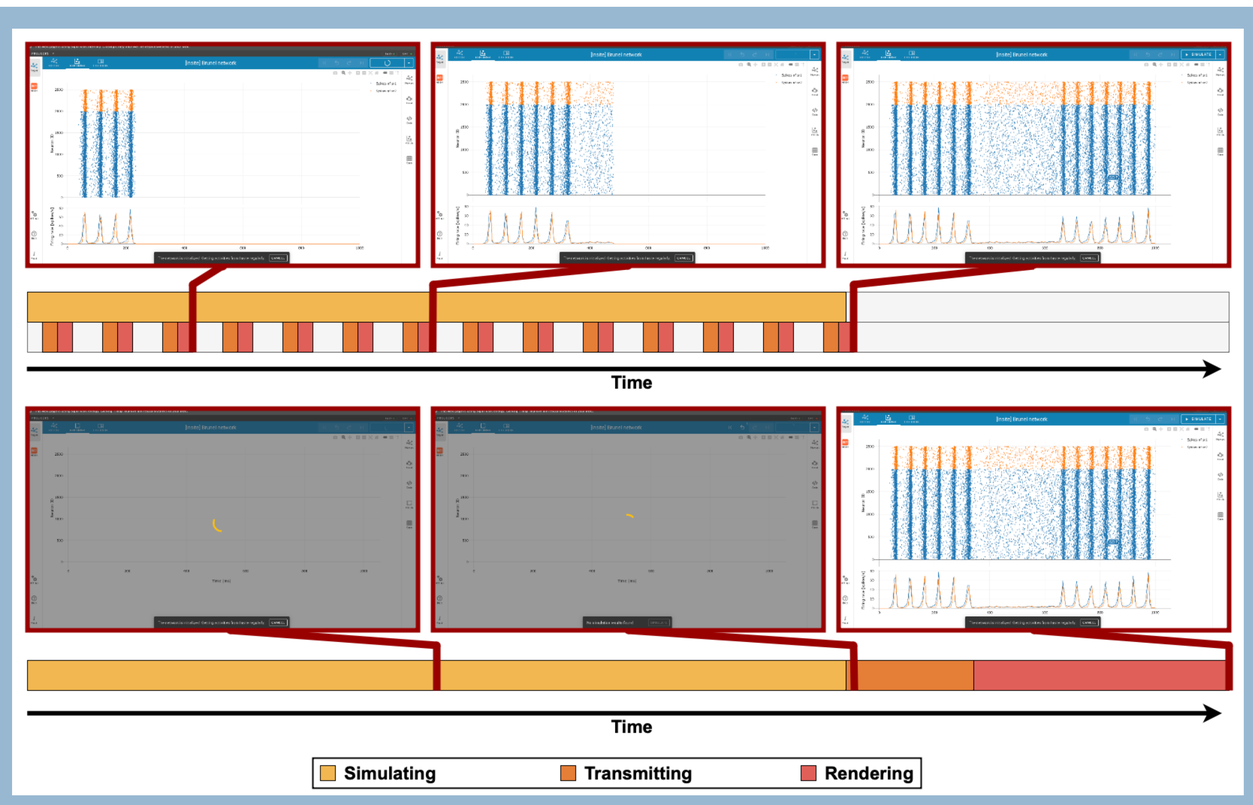

Figure 3. NEST Desktop with (top) and without (bottom) Insite Pipeline. With Insite Pipeline the user receives feedback as soon as data is available. Transmitting and Rendering is interleaved. Without the user has to wait until the simulation finished, the data is transmitted and rendered.

[1] Krüger et al. (2022). Insite: A Pipeline Enabling In-Transit Visualization and Analysis for Neuronal Network Simulations. In: Anzt et al. (eds) High Performance Computing. ISC High Performance 2022 International Workshops. Lecture Notes in Comp. Sc., vol 13387. Springer, Cham.
[2] Oehrl et al. (2018). Streaming Live Neuronal Simulation Data into Visualization and Analysis. In: Yokota et al. (eds) High Performance Computing. ISC High Performance 2018. Lecture Notes in Comp. Sc., vol 11203. Springer, Cham.
[3] Galindo et al. (2016). ViSimpl: multi-view visual analysis of brain simulation data. Frontiers in Neuroinformatics, 10, 44.
[4] Gewaltig, & Diesmann (2007). NEST (NEural Simulation Tool). Scholarpedia, 2(4), 1430.
[5] Abi Akar et al. (2022). Arbor Library v0.8.1 (v0.8.1). Zenodo.
[6] Sanz Leon et al. (2013). The Virtual Brain: a simulator of primate brain network dynamics, Frontiers in Neuroinformatics 7:10.
[7] Buchertseifer et al. (2012). Combining in-situ and in-transit processing to enable extreme-scale scientific analysis. In SC'12: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis (pp. 1-9). IEEE.

GitHub

Marcel Krüger

Tel.: +49 (0)241 80 - 24966
E-Mail: krueger@vis.rwth-aachen.de

EBRAINS

Co-funded by the European Union

This research has received funding from the European Union's Horizon 2020 Framework Programme for Research and Innovation under the Specific Grant Agreement No. 945539 (Human Brain Project SGA3).